https://ntrs.nasa.gov/search.jsp?R=19820008874 2020-03-21T10:40:38+00:00Z

N O T I C E

THIS DOCUMENT HAS BEEN REPRODUCED FROM MICROFICHE. ALTHOUGH IT IS RECOGNIZED THAT CERTAIN PORTIONS ARE ILLEGIBLE, IT IS BEING RELEASED IN THE INTEREST OF MAKING AVAILABLE AS MUCH INFORMATION AS POSSIBLE

NASA CR-165539
TRW 32660-6001-RU-01

MODELING AND ANALYSIS OF POWER PROCESSING SYSTEMS (MAPPS)

FINAL REPORT

VOLUME II - APPENDICES

By

Dr. F. C. Lee, Dr. S. Rahman,
R. A. Carter, C. H. Wu

VIRGINIA POLYTECHNIC
INSTITUTE & STATE UNIVERSITY

DR. YUAN YU
R. CHANG

TRW DEFENSE & SPACE
SYSTEMS GROUP

Prepared for:

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

LEWIS RESEARCH CENTER
21000 Brookpark Road
Cleveland, Ohio 44135

CONTRACT NAS3-21051

(NASA-CR-165539) MODELING AND ANALYSIS OF POWER PROCESSING SYSTEMS (MAPPS). VOLUME 2: APPENDICES Final Technical Report, Oct. 1977 - Aug. 1980 (Virginia Polytechnic Inst. and State Univ.) 210 p HC A10/MF A01 G3/60

N82-16748

Unclas
05538

NASA CR-165539
TRW 32660-6001-RU-01

MODELING AND ANALYSIS
OF
POWER PROCESSING SYSTEMS (MAPPS)

FINAL REPORT

VOLUME II - APPENDICES

By

Dr. F. C. Lee, Dr. S. Rahman,
R. A. Carter, C. H. Wu

VIRGINIA POLYTECHNIC
INSTITUTE & STATE UNIVERSITY

DR. YUAN YU
R. CHANG

TRW DEFENSE & SPACE
SYSTEMS GROUP

Prepared for:

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

LEWIS RESEARCH CENTER
21000 Brookpark Road
Cleveland, Ohio 44135

CONTRACT NAS3-21051

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
NAS CR-165539		

4. Title and Subtitle	5. Report Date
MODELING AND ANALYSIS OF POWER PROCESSING SYSTEMS (MAPPS), VOLUME II - APPENDICES	DECEMBER 1980
	6. Performing Organization Code: TRW 32660-6001-RU-01
7. Author(s): F. C. Lee, S. Radman, R. A. Carter, C. H. Wu, Yuan Yu, and R. Chang	8. Performing Organization Report No.
	10. Work Unit No.
9. Performing Organization Name and Address: TRW Defense and Space Systems Group, Power Conversion Electronics Department, One Space Park, Redondo Beach, California 90273	11. Contract or Grant No.: NAS 3-21051
	13. Type of Report and Period Covered: FINAL TECHNICAL REPORT OCT. 1977-AUG. 1980
12. Sponsoring Agency Name and Address: NASA Lewis Research Center, 21000 Brookpark Road, Cleveland, Ohio 44135	14. Sponsoring Agency Code

15. Supplementary Notes

NASA Technical Monitor: Mr. Joseph Kolecki

16. Abstract

Volume I covers the technical results of the Modeling and Analysis of Power Processing Systems.

Volume II provides the computer programs and derivations that were generated in support of the technical contract work.

17. Key Words (Suggested by Author(s))	18. Distribution Statement
Modeling & Analysis, Buck-Boost Regulator, Power Processor, Performance Design, DC-DC Converter, Design Optimization, Buck Regulator, Converter Aided Design, Boost Regulator,	UNCLASSIFIED - UNLIMITED

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price*
UNCLASSIFIED		210	

*For sale by the Clearinghouse for Federal Scientific and Technical Information, Springfield, Virginia 22151

NASA-168

FOREWORD

The Modeling and Analysis tasks were performed by the following personnel:

Task I - Discrete Time Domain Analysis of Switching Regulator

Dr. Yuan Yu and Ron Chang
TRW Defense and Space Systems Group

Task II - Design Optimization of Power Converters

Dr. F. C. Lee, Dr. S. Rahman, C. H. Wu
Virginia Polytechnic Institute & State University

Task III - Investigation of Current Injected Multiloop Controlled Switching Regulators

Dr. F. C. Lee and R. A. Carter
Virginia Polytechnic Institute & State University

Task IV - MAPPS Demonstration Problem for VSTOL Emergency Power Systems

Dr. Yuan Yu
TRW Defense and Space Systems Group

The authors wish to acknowledge the contribution to this work by the NASA Project Monitor, Mr. Joseph Kolecki, whose thorough review, numerous comments and suggestions helped to improve this final report.

TABLE OF CONTENTS

INTRODUCTION

The following appendices documents pertinent supporting technical effort that was performed for the Modeling and Analysis program:

Appendices A through C contains the computer program listings that were generated as part of Task I, "Discrete Time Domain Analysis of Switching Regulators."

Appendix A contains the software for the Buck Regulator, Appendix B contains the software for the Boost Regulator, and Appendix C contains the software for the Buck-Boost Regulator. All programs have been checked out and the results compared with typical performance data. Problem areas may still exist when selecting the error coefficient value (EPS) used in the programs.

Appendices D through G contains the equation derivation and optimization software for Boost and Buck-Boost Regulators developed in support of Task II, "Design Optimization of Power Converters." Software has been checked out.

Appendices H and I were performed as part of Task II, "Investigation of Current Injected Multiloop Controlled Switching Regulators."

Appendix J contains the computer program developed and used for Task IV, "MAPPS Demonstration Problem for VSTOL Emerging Power System."

Appendix K establishes the input parameter justification for the discussion presented in Section 5, Volume I.

Appendix A. Buck Regulator PAS Computer Program Description

The computer program nomenclature is documented in Tables A1 through A3.

The PAS1 computer program flow chart is presented in Figures A1 through A3. Figure A1 depicts the computational flow for the various analysis techniques. Figures A2 and A3 represent the user interactive guide line for directing the computational flow of the program. The flow charts presented in Figures A4 through A13 describe the computational sequences of the various subroutines and functions that comprise the sumary functioned blocks of the buck regulator program.

In addition, the derivation of the computational procedure for the closed-loop transfer function for the audio susceptibility analyses is presented in Figure A14.

Table A1. PAS1 Computer Program Nomenclature

RIPX(3,1)	steady-state ripple = $\underline{x} - \underline{y}$
PSI(3,3)	Ψ
PSY(3,3)	Ψ (for eigenvalue computations)
GAM(3,1)	Γ
INT(8)	QRAL library subroutine
XFP(3,1)	$\underline{x}_{OLD}$
PRAM(10)	[C1 C2 R3 R4 R5 RN2 XLO CO RL EI]^T
H(3)	H
R(3,5)	QRAL library subroutine
ITBL(3)	QRAL library subroutine
IVD(3)	QRAL library subroutine
DELX(3,1)	$\Delta\underline{x}$
PHI1(3,3)	$\Phi(T_{ON})$
PHI2(3,3)	$\Phi(T_{F1})$
PHI3(3,3)	$\Phi(T_{F2})$
D1(3,2)	$D(T_{ON})$
D2(3,2)	$D(T_{F1})$
D3(3,2)	$D(T_{F2})$
TEMP1(3,3)	dummy matrix
TVEC1(3)	dummy vector
PHIP(3,3)	$\Phi(T_p)$

Table A1. PAS1 Computer Program Nomenclature (Continued)

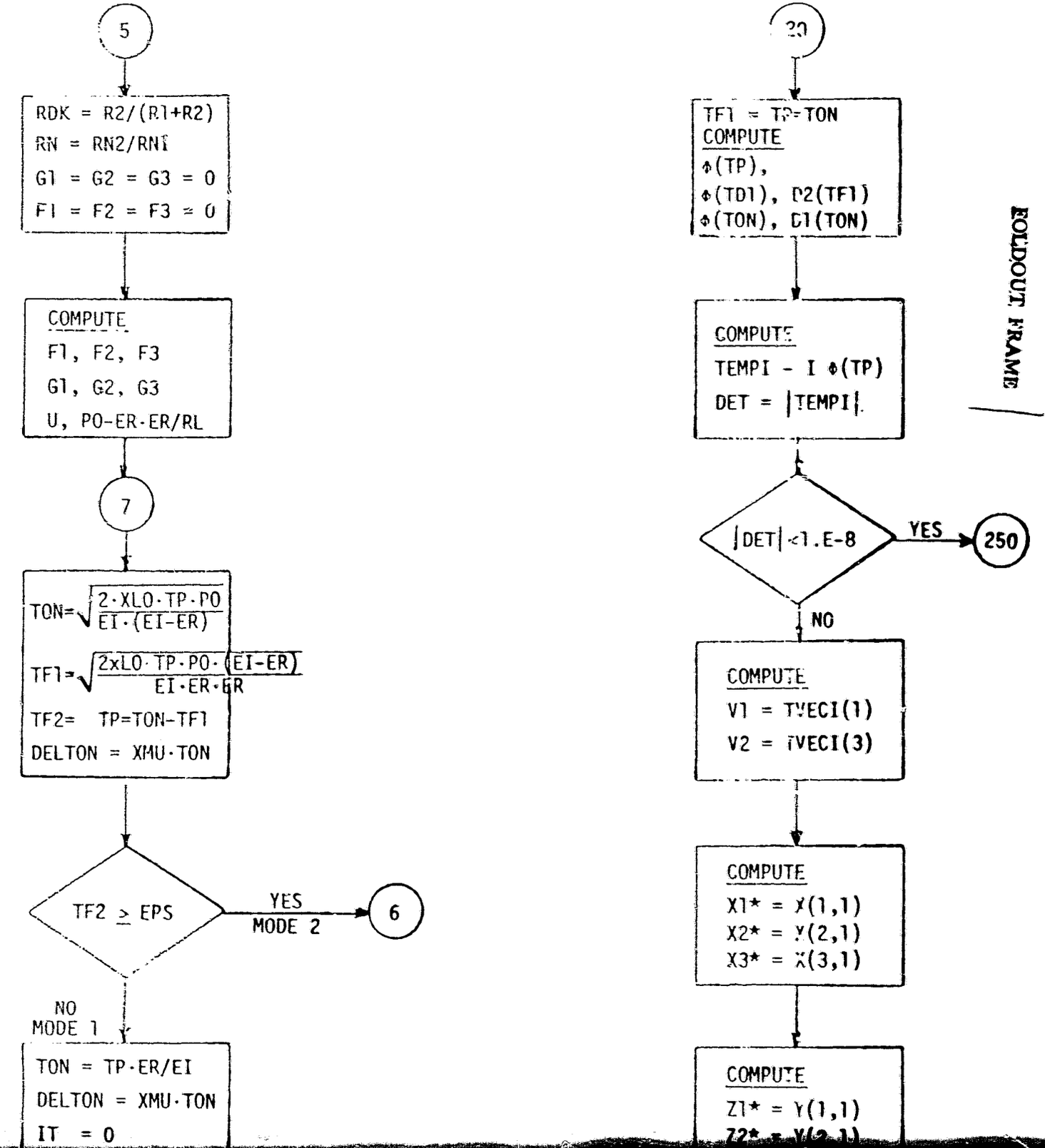
COMMON/PARAM/	
F1(3,3)	state eqns characteristic matrix
F2(3,3)	state eqns characteristic matrix
F3(3,3)	state eqns characteristic matrix
G1(3,2)	state eqns input coupling matrix
G2(3,2)	state eqns input coupling matrix
G3(3,2)	state eqns input coupling matrix
COMMON/EXTPAR/	
NIT	maximum number iterations for steady-state solution
EPS	minimum error for steady-state solutions
TP	T_p
ET	E_T
MODE	inductor current flag { = 1 - continuous; = 2 - discontinuous
COMMON/STATE/	
X(3,1)	$\underline{x}$
Y(3,1)	$\underline{y}$
Z(3,1)	$\underline{z}$
U(2,1)	$\underline{u}$

Table A2. Nominal Parameters (MKS System)

EI	50
ER	20
ET	8
RL	10
TP	30E-6
EISWIT	60.
XMU	0.01
XLO	25.E-5
RO	0.015
CO	3.E-4
R5	0.077
RN1	40.
RN2	26.
C1	2200E-12
C2	0.022E-6
R1	28.7E-3
R2	13.5E-3
R3	10.E-3
R4	100.E-3
THETAO	0
DELTHET	5.
THETAF	180.
H	$[1 \quad 0 \quad 0]^T$
NIT	100
EPS	1.E-6
MODE	2
IPLØT	0
LIST	0
LPEAK	0
LFE	0
NK	15
LFREQ	0
LRTL	0
NRL	2
DPRAM	0
PRAMF	0

Table A3. Namelist Parameters

NAMELIST/PARAM/	NAMELIST/COMP/
EI	EPS
ER	NIT
ET	XMU
RL	
TP	
XLO	
RO	
CO	
R5	
RN1	
RN2	
C1	
C2	
R1	
R2	
R3	
R4	



FOLDOUT FRAME

ORIGINAL PAGE IS
OF POOR QUALITY

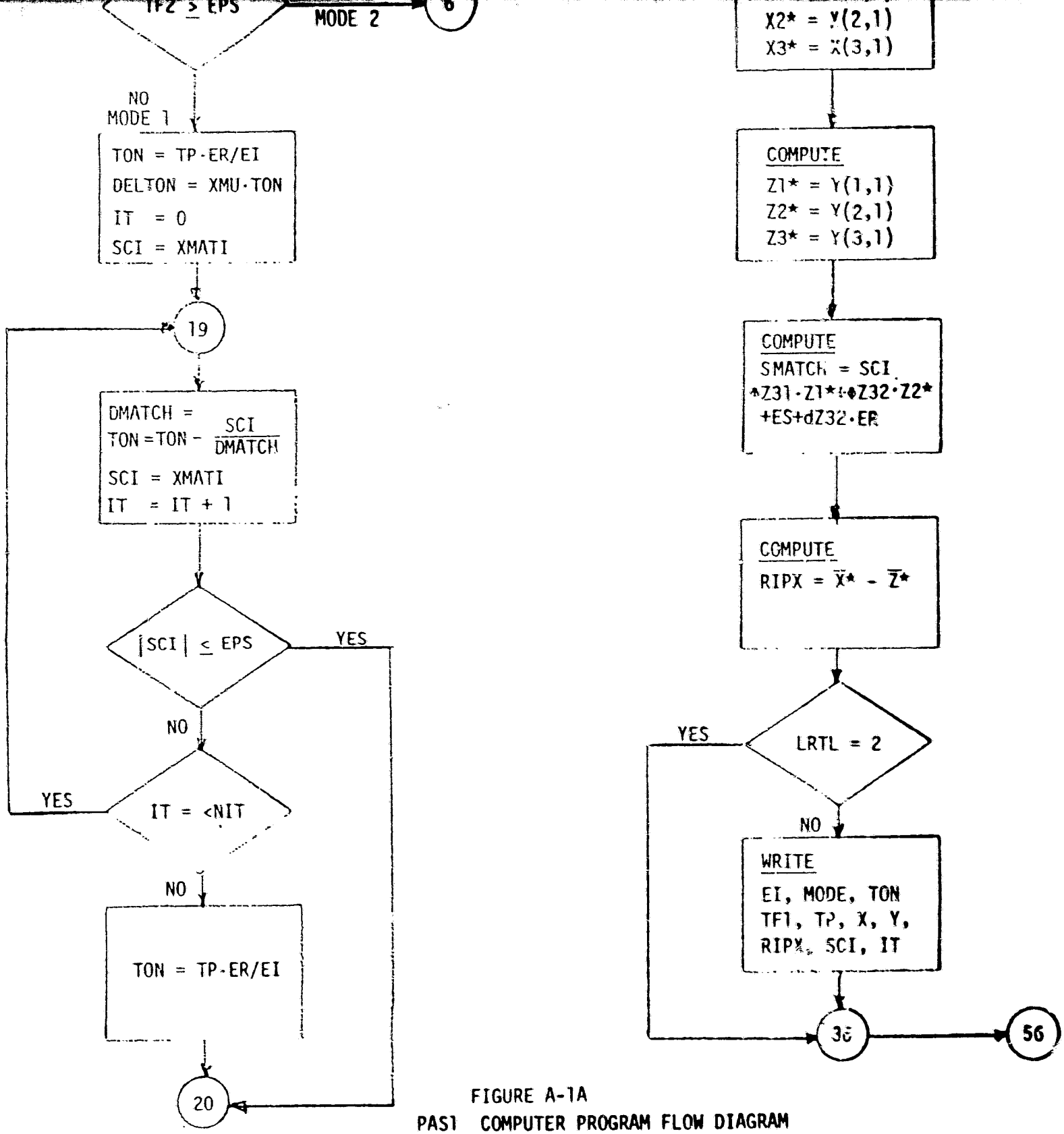
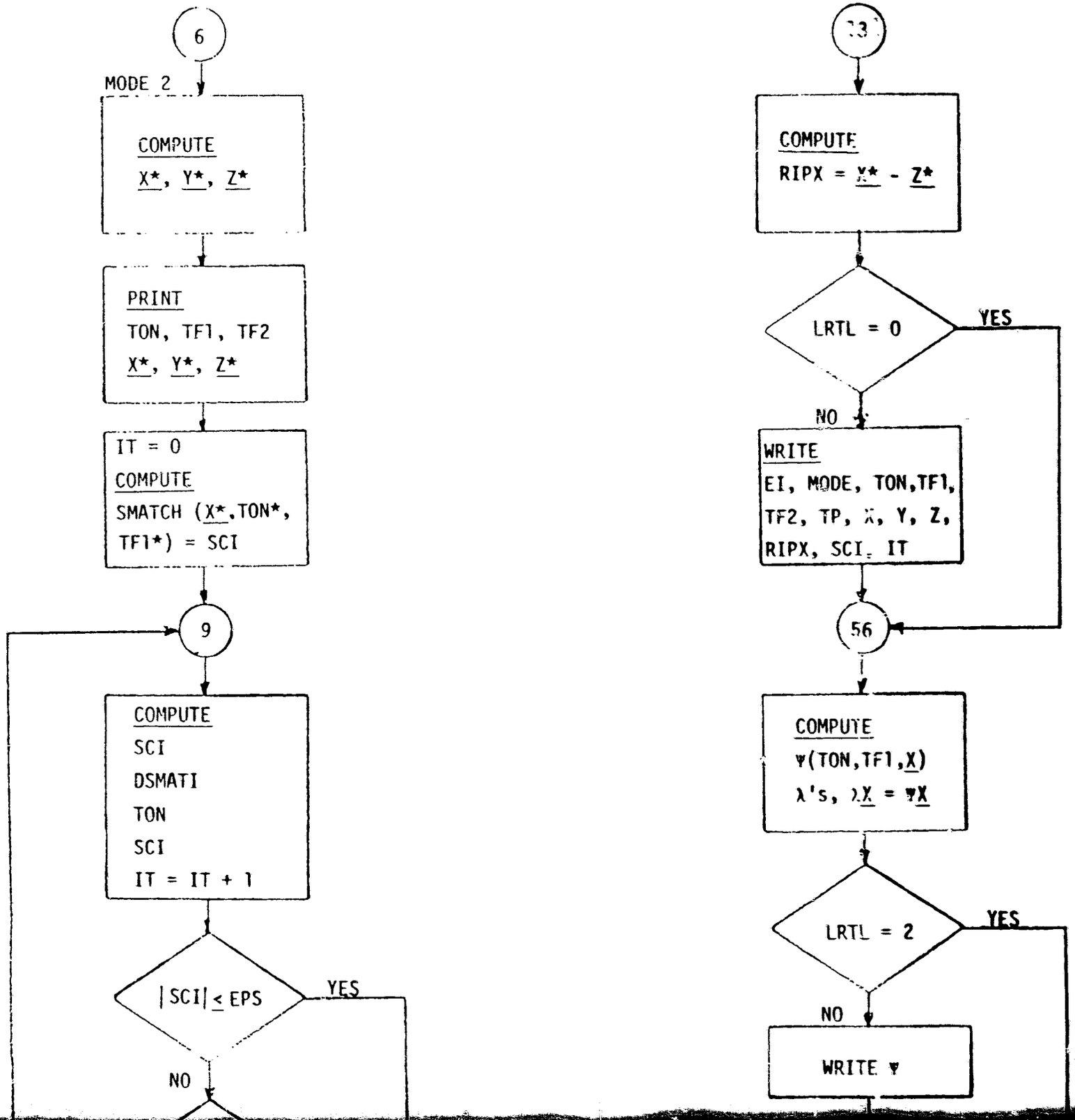


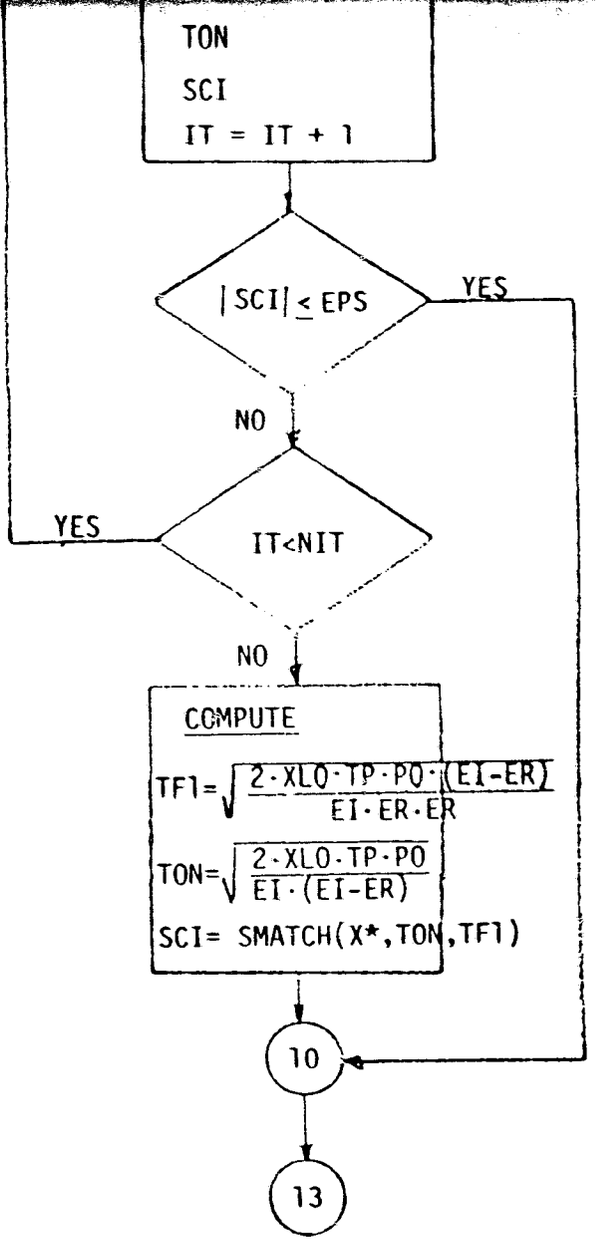
FIGURE A-1A
PAS1 COMPUTER PROGRAM FLOW DIAGRAM

ORIGINAL PAGE IS
OF POOR QUALITY

ROLLOUT FRAME 2



FOLDOUT FRAME



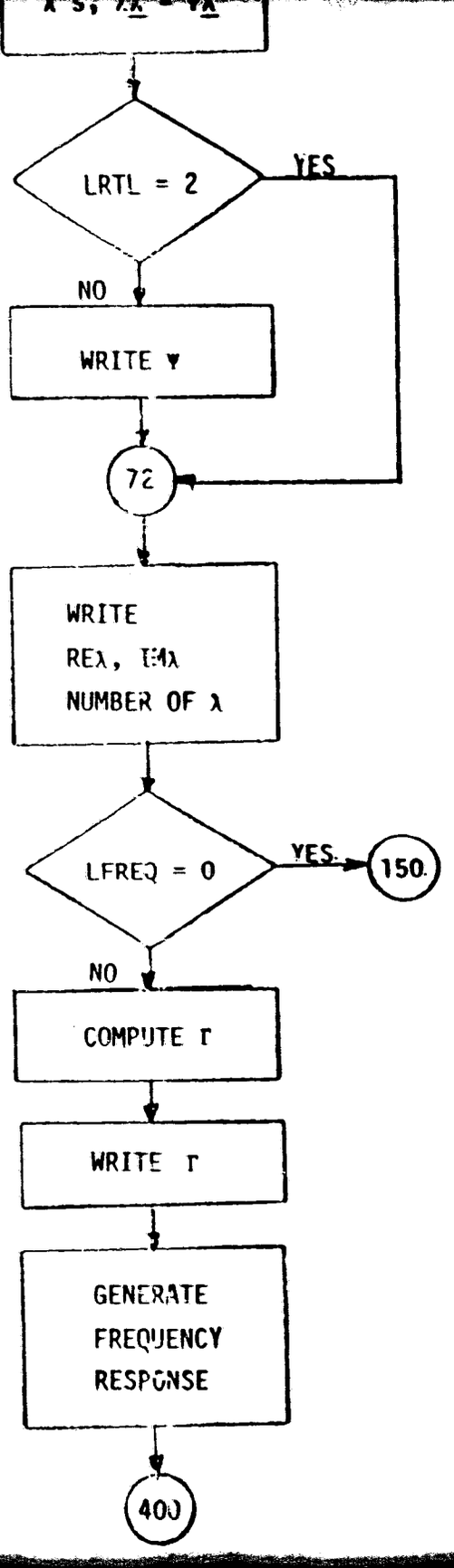
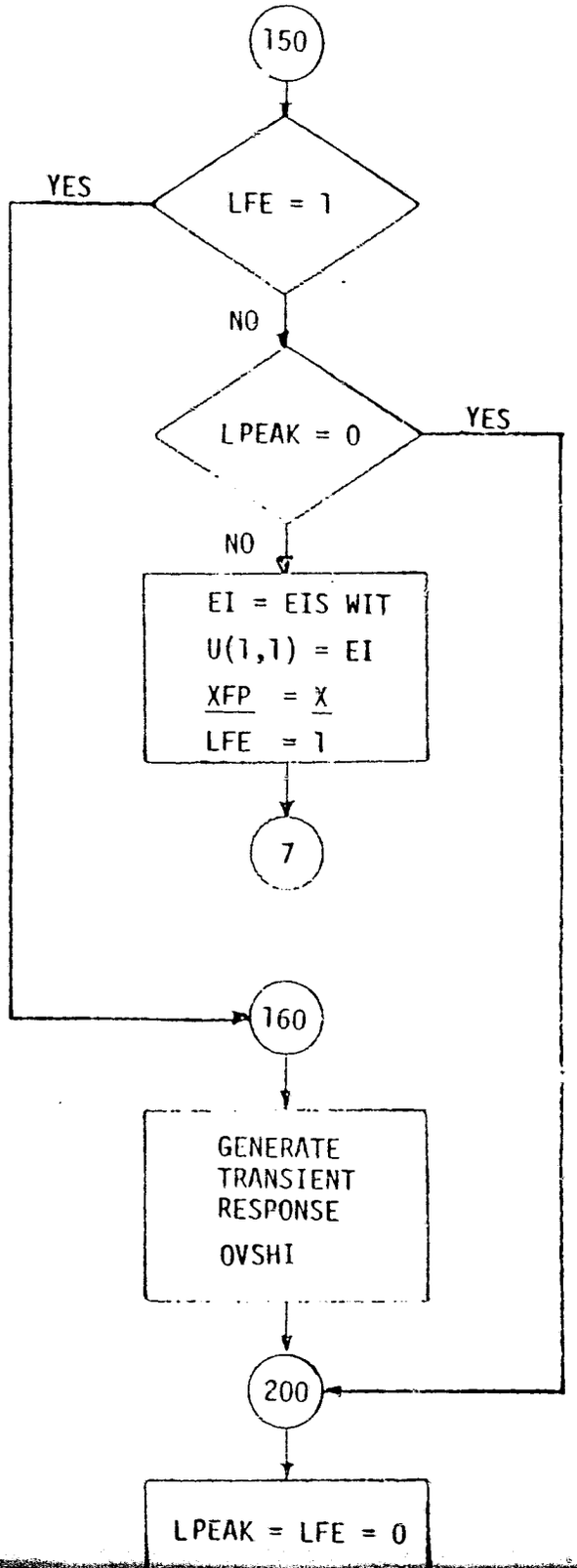
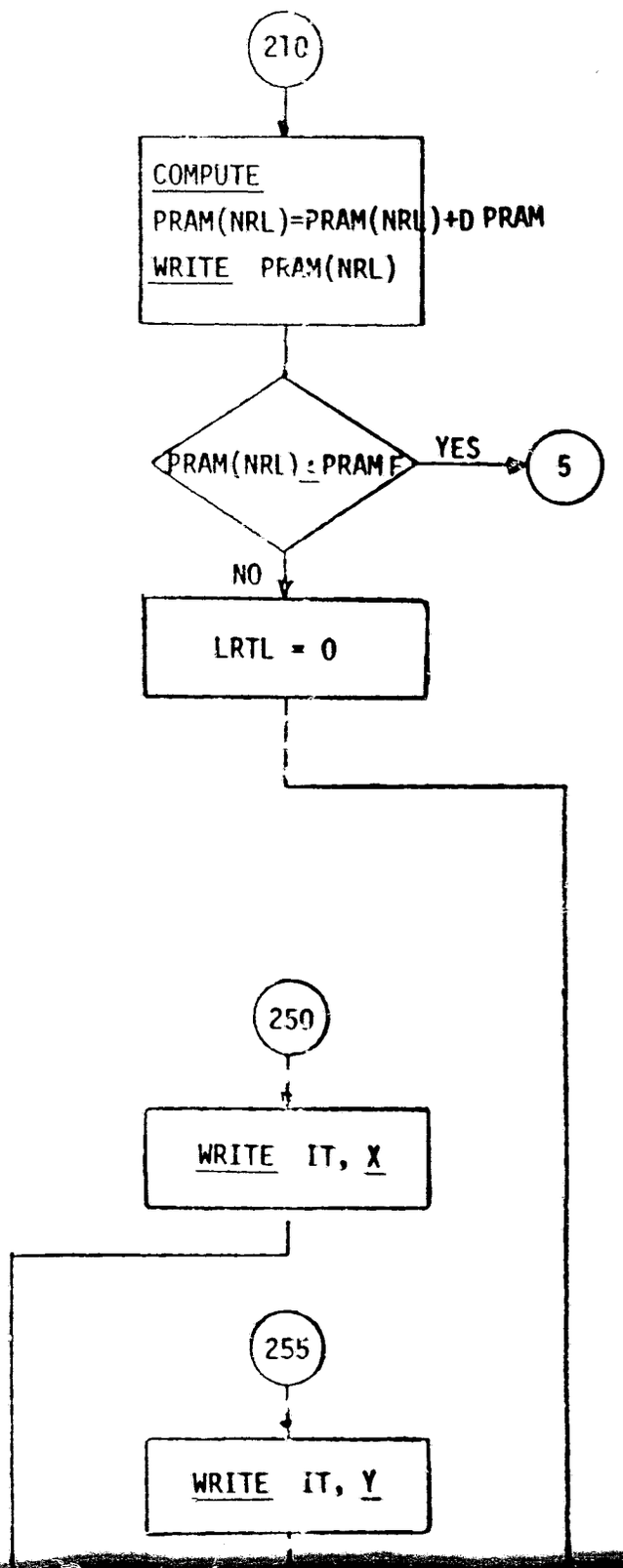


FIGURE A-1B - PAS 1 COMPUTER FLOW CHART

FOLDOUT FRAME

FOLDOUT FRAME





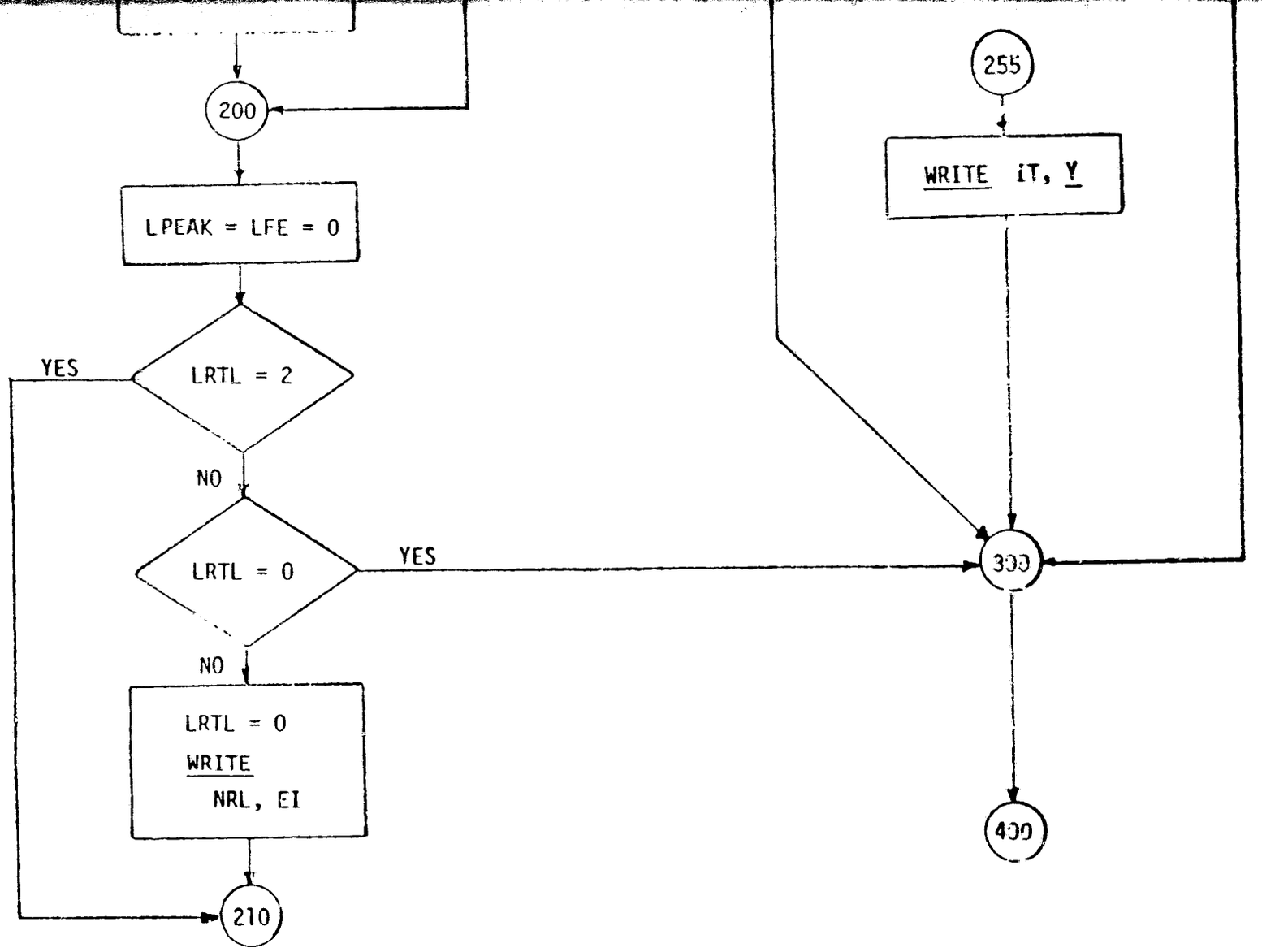
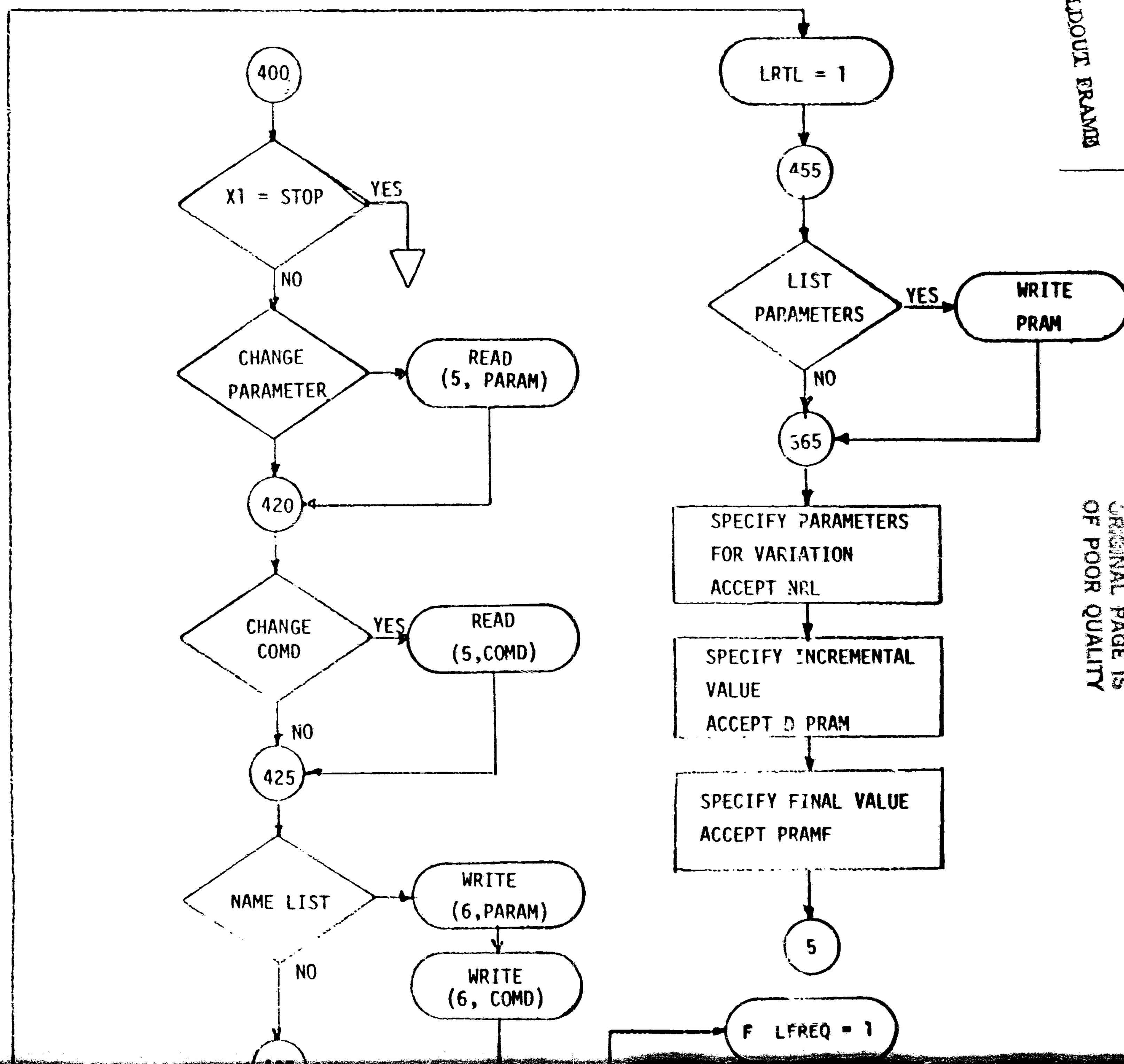


FIGURE A-1C

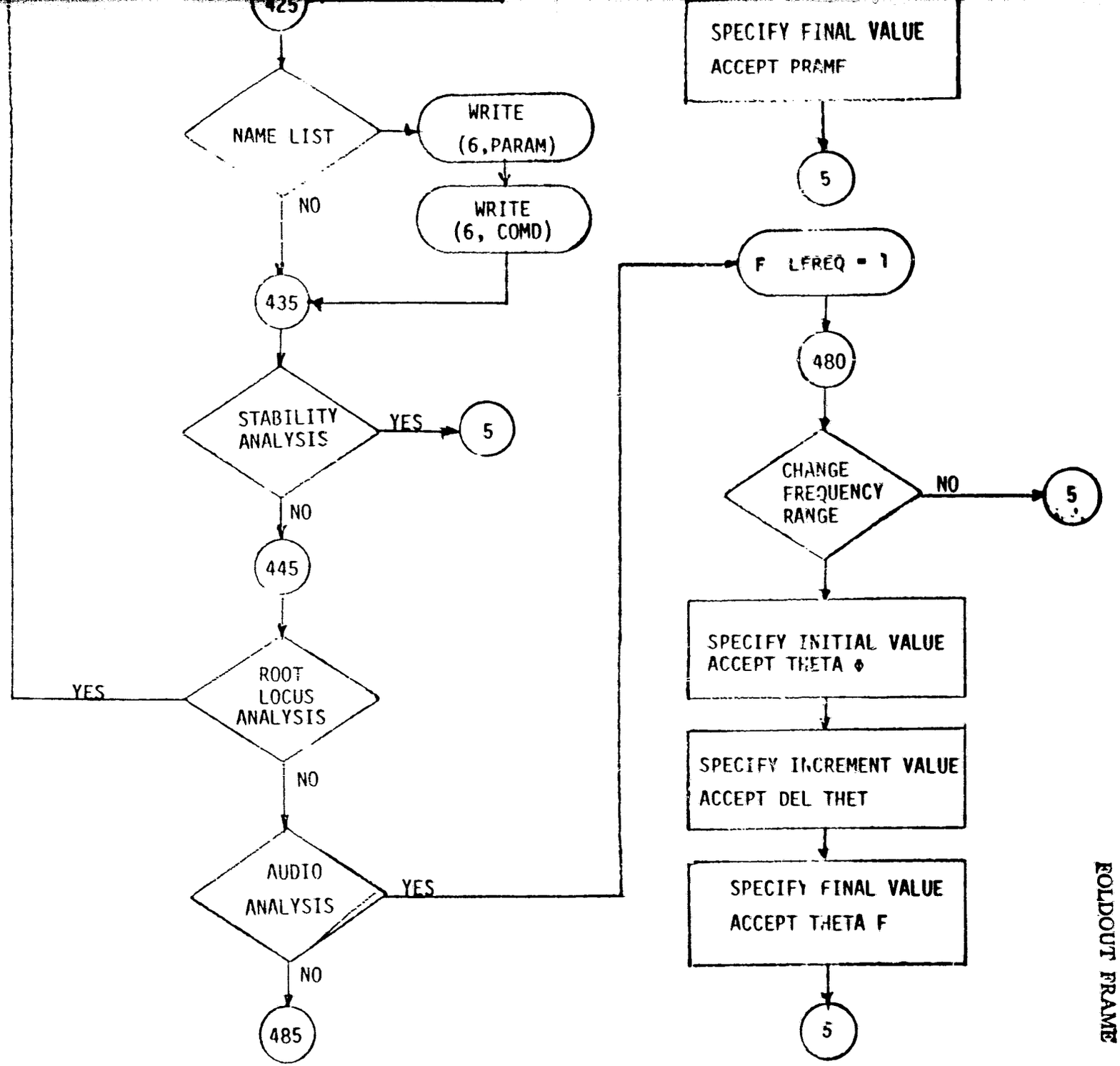
PAS1 COMPUTER FLOW CHART

FOLDOUT FRAME



FOLDOUT FRAME

ORIGINAL PAGE IS OF POOR QUALITY



FOLDOUT FRAME 2

FIGURE A2 - PAS 1 COMPUTER PROGRAM USER

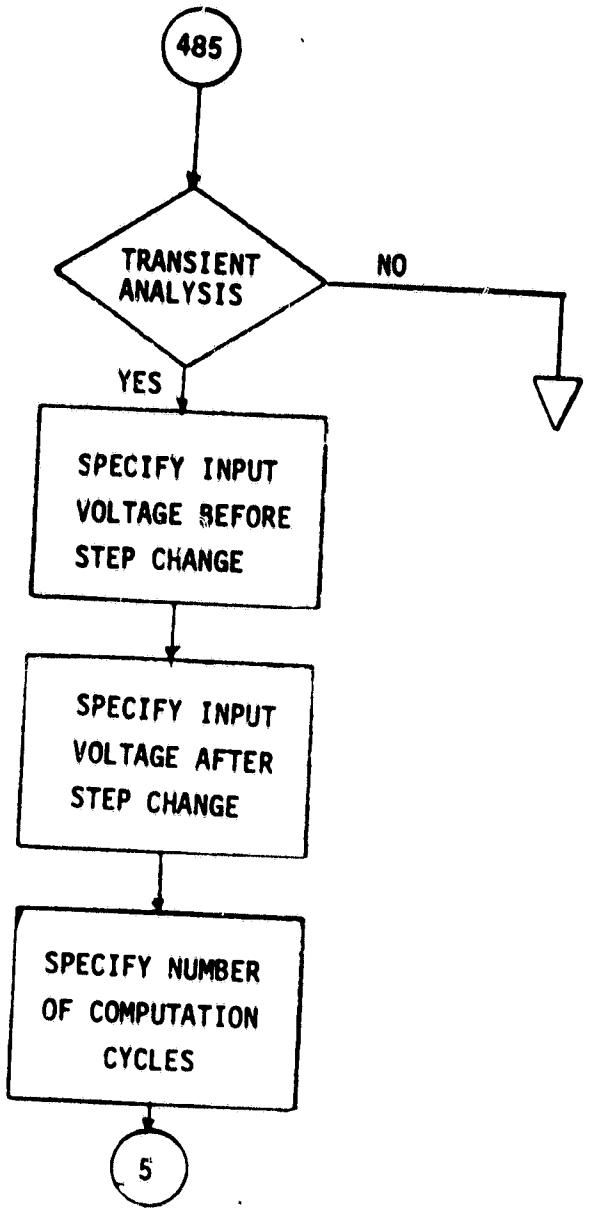
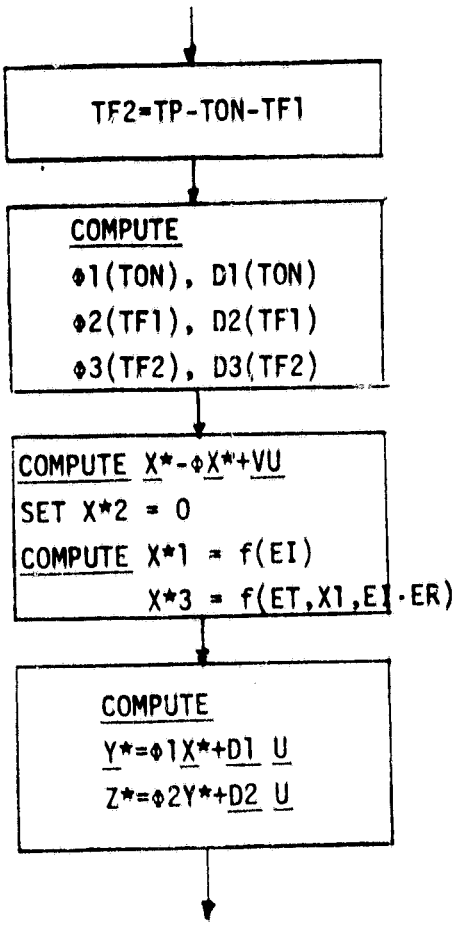


FIGURE A3 - PAS 1 COMPUTER PROGRAM USER INTERFACE GUIDE

SUBROUTINE STATE 1 (TON, TF1)



SUBROUTINE STS 1 (W2, PHI, W1, D, U)

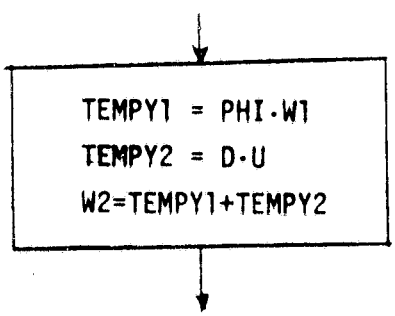
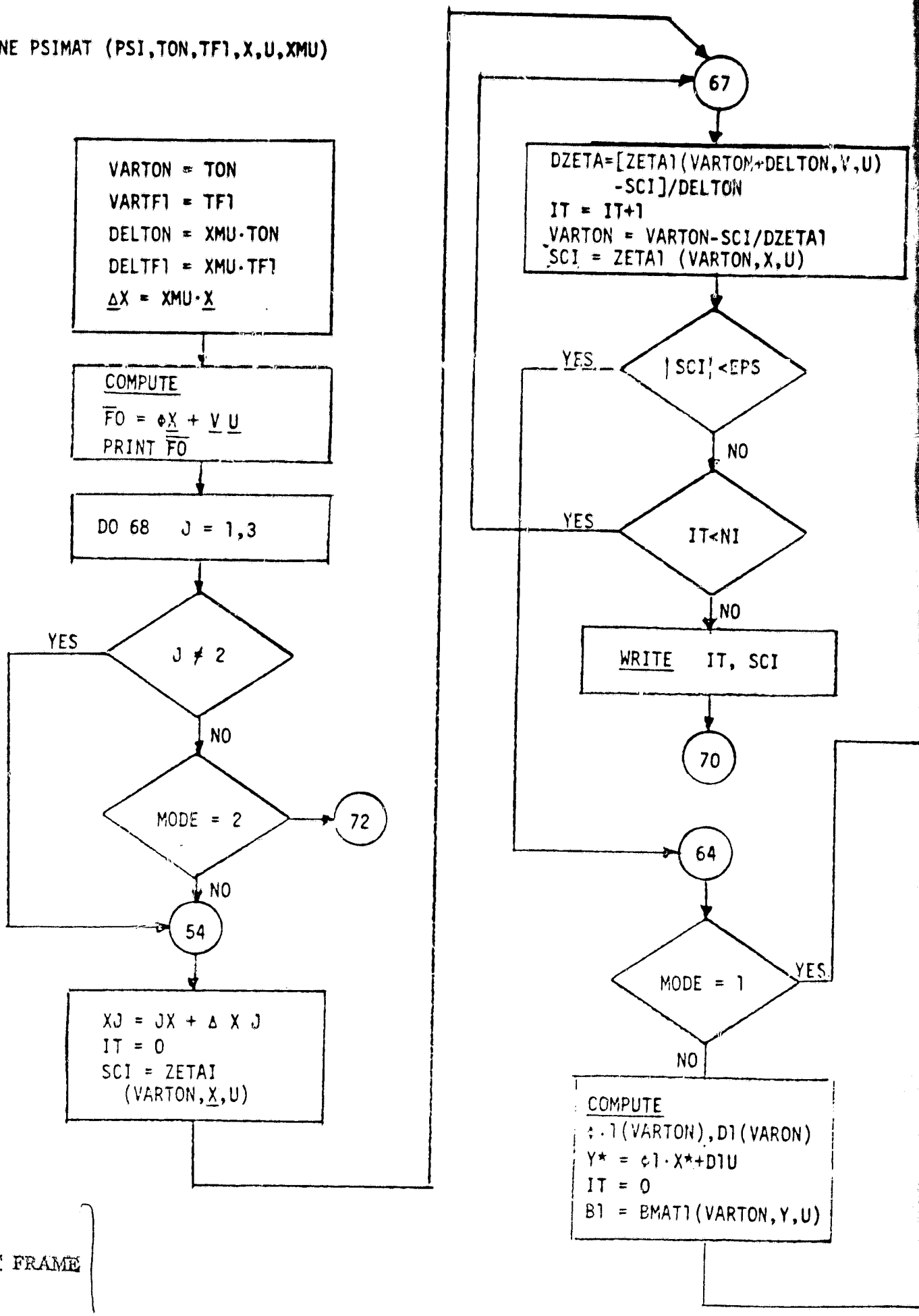
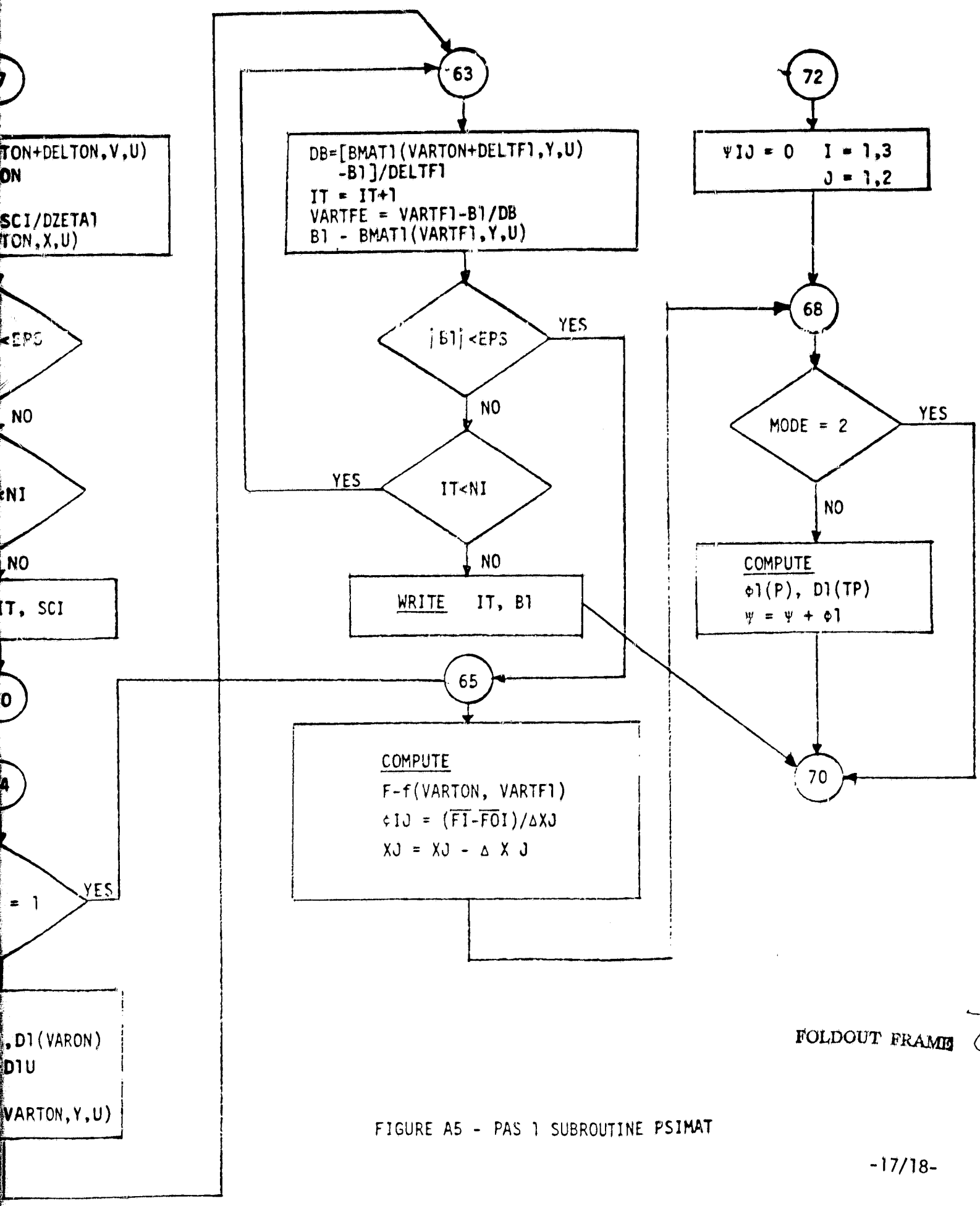


FIGURE A4 - PAS 1 SUBROUTINES STATE 1 and STS 1

SUBROUTINE PSIMAT (PSI,TON,TF1,X,U,XMU)



FOLDOUT FRAME



FOLDOUT FRAME 2

FIGURE A5 - PAS 1 SUBROUTINE PSIMAT

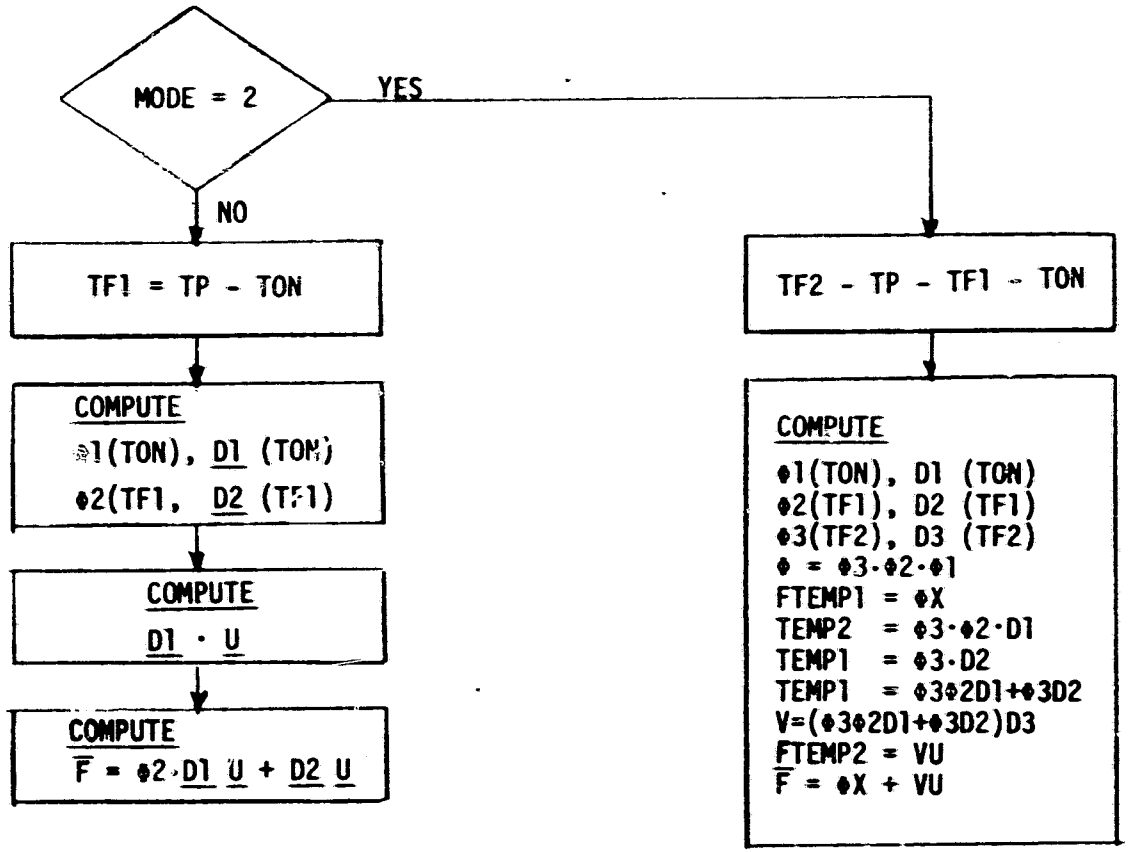


FIGURE A6 - PAS 1 SUBROUTINE FFUNC 1

SUBROUTINE GAMM 1 (GAM, TON, TF1, X, U, XMU)

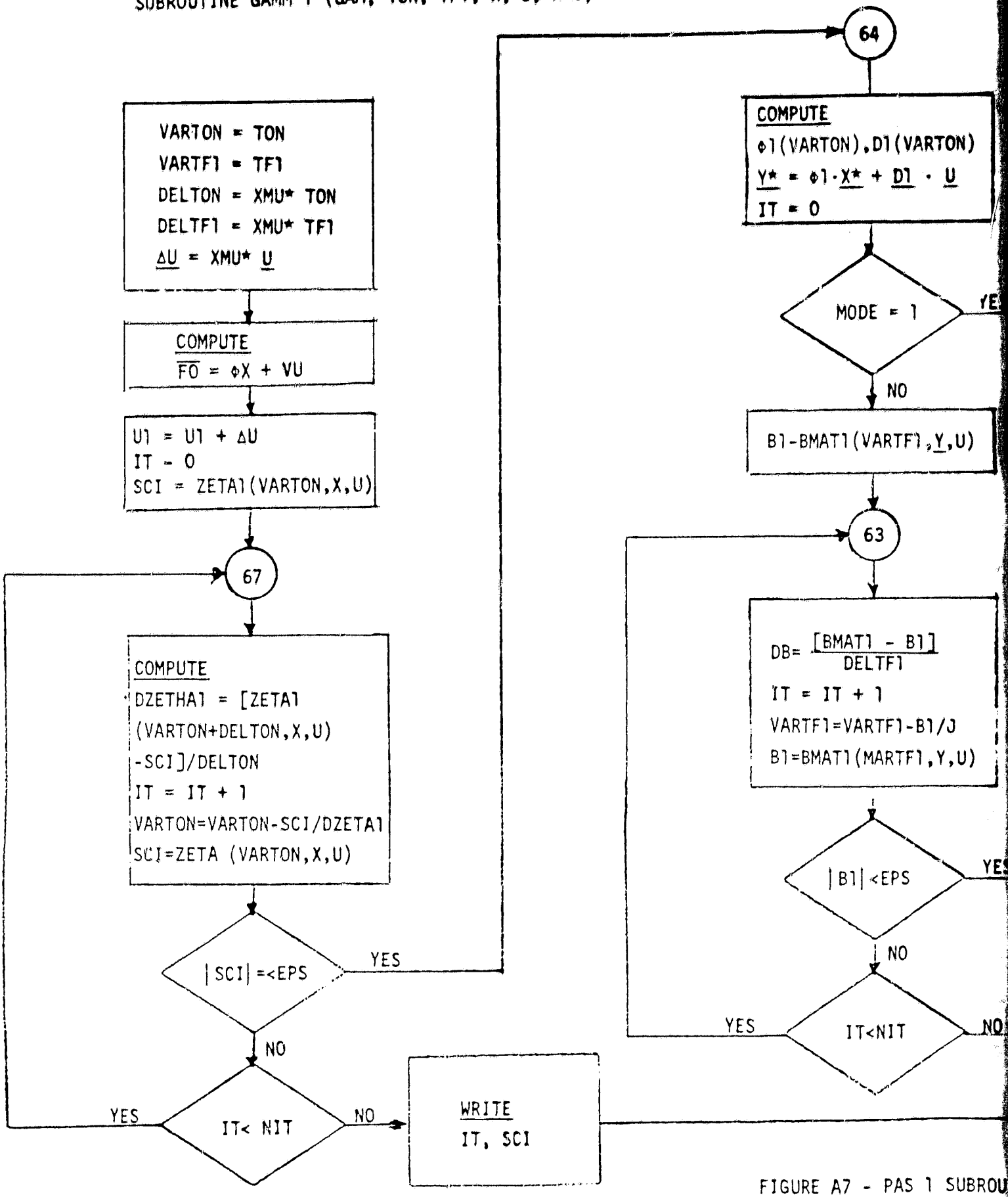


FIGURE A7 - PAS 1 SUBROU

FOLDOUT FRAME

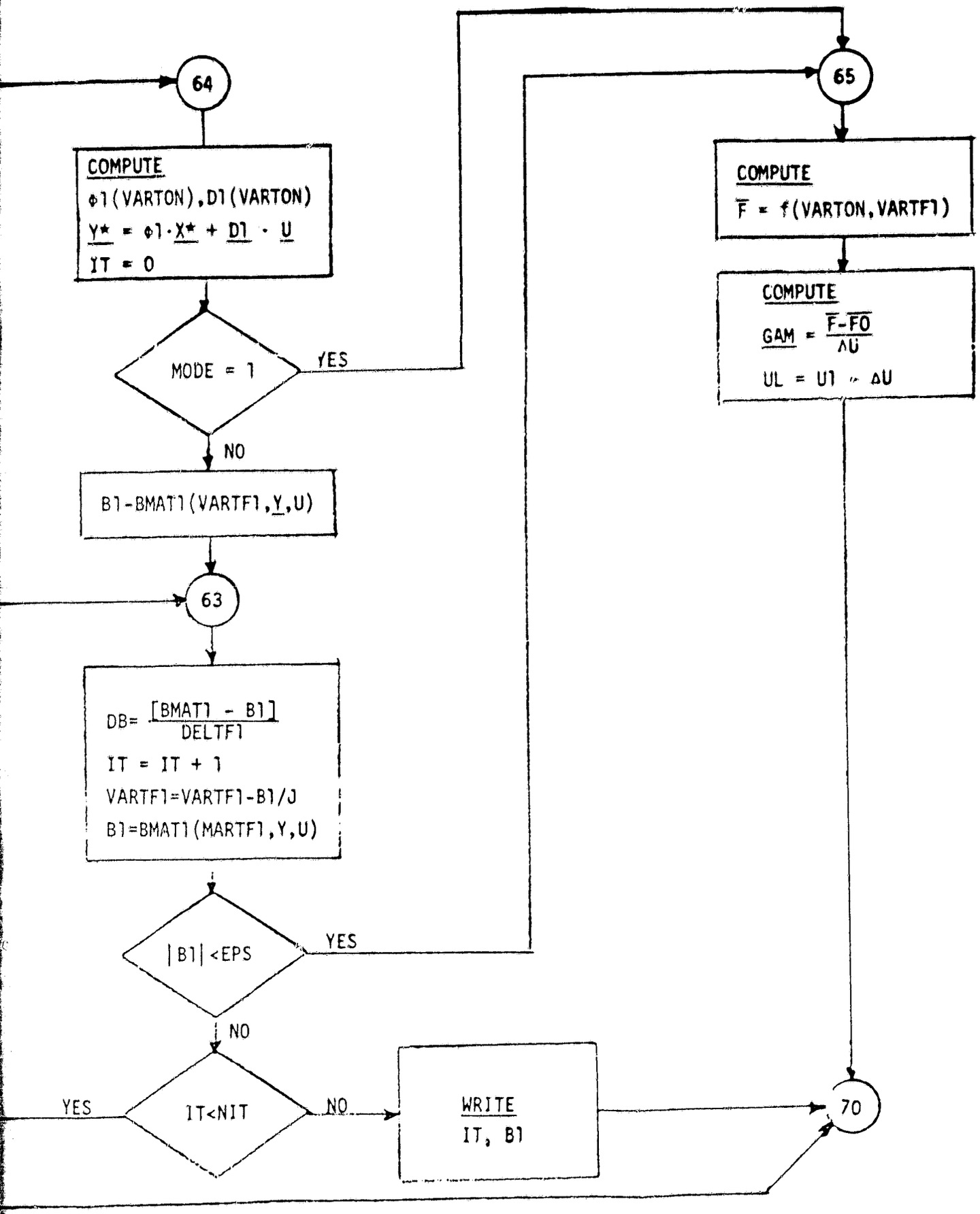


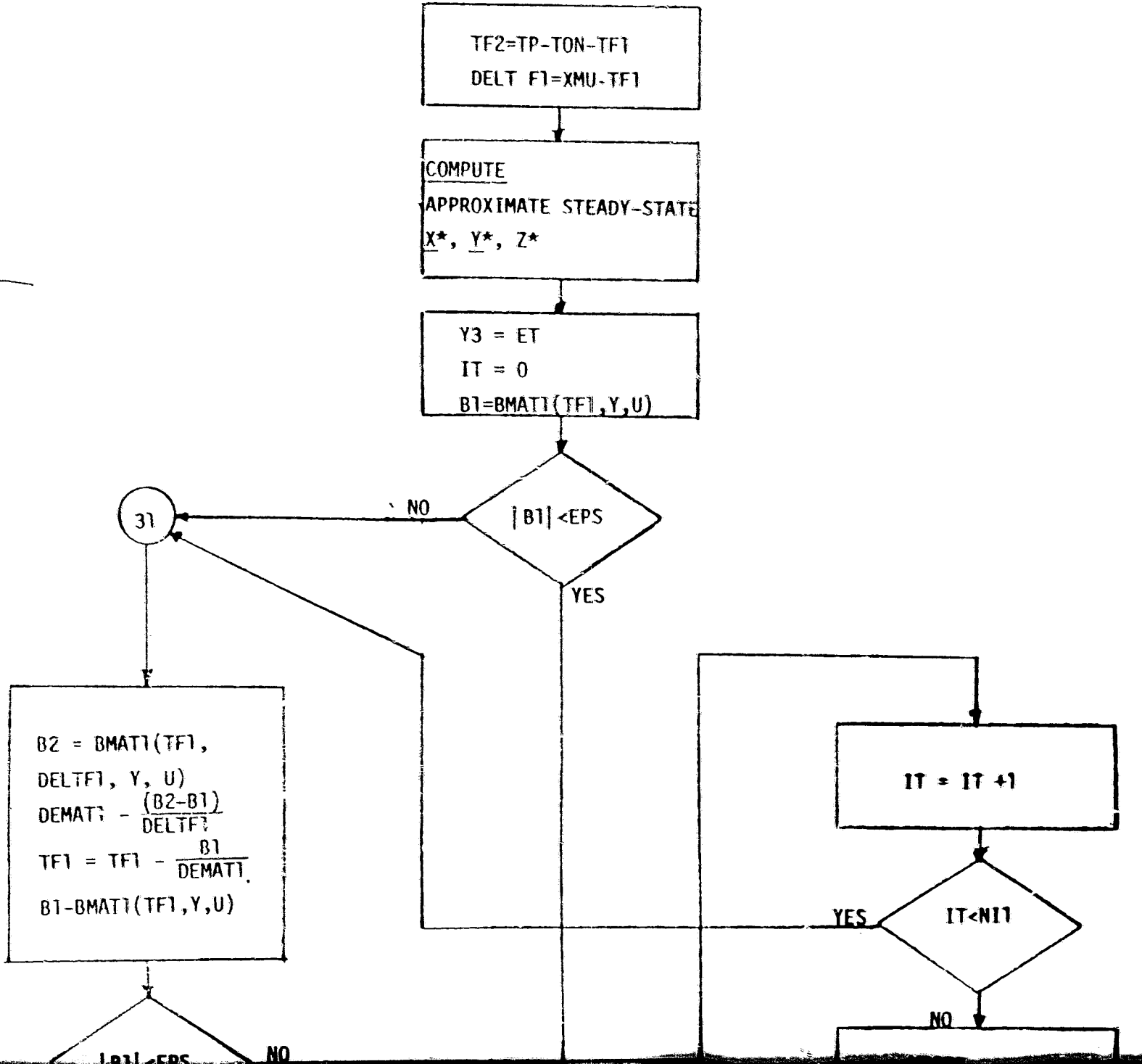
FIGURE A7 - PAS 1 SUBROUTINE GAMM 1

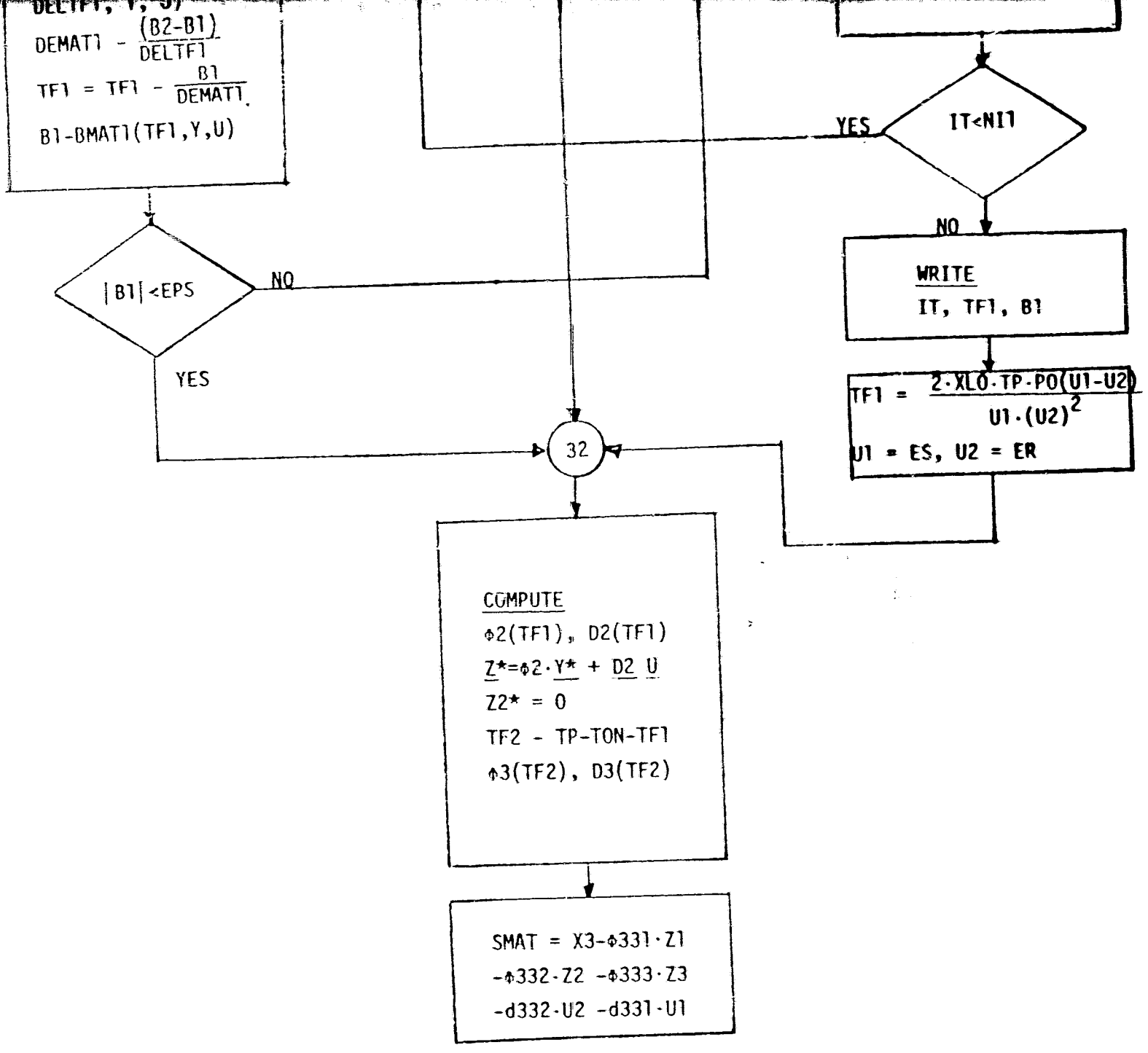
FOLDOUT FRAME

FOLDOUT FRAME

FOLDOUT FRAME

SUBROUTINE SMAT1 (TON,TF1,TF2, SMAT, XMU, XLO, PC)

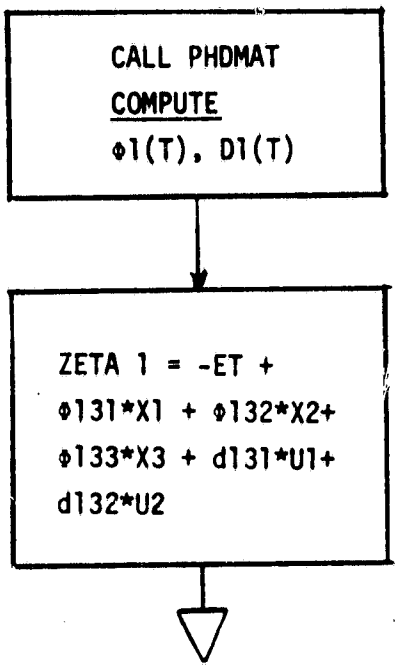




FOLDOUT FRAME 2

FIGURE A8 - PAS 1 SUBROUTINES SMAT 1

SUBROUTINE ZETA 1 (T, X, U)



SUBROUTINE BMAT 1 (TF 1, Y, U)

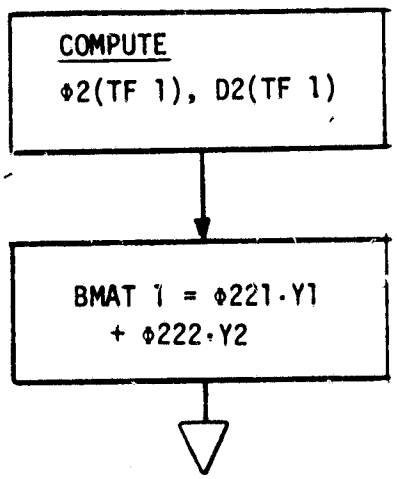


FIGURE A9 - PAS 1 SUBROUTINE ZETA 1 AND BMAT 1

SUBROUTINE XMAT1 (TON, EI, ER)

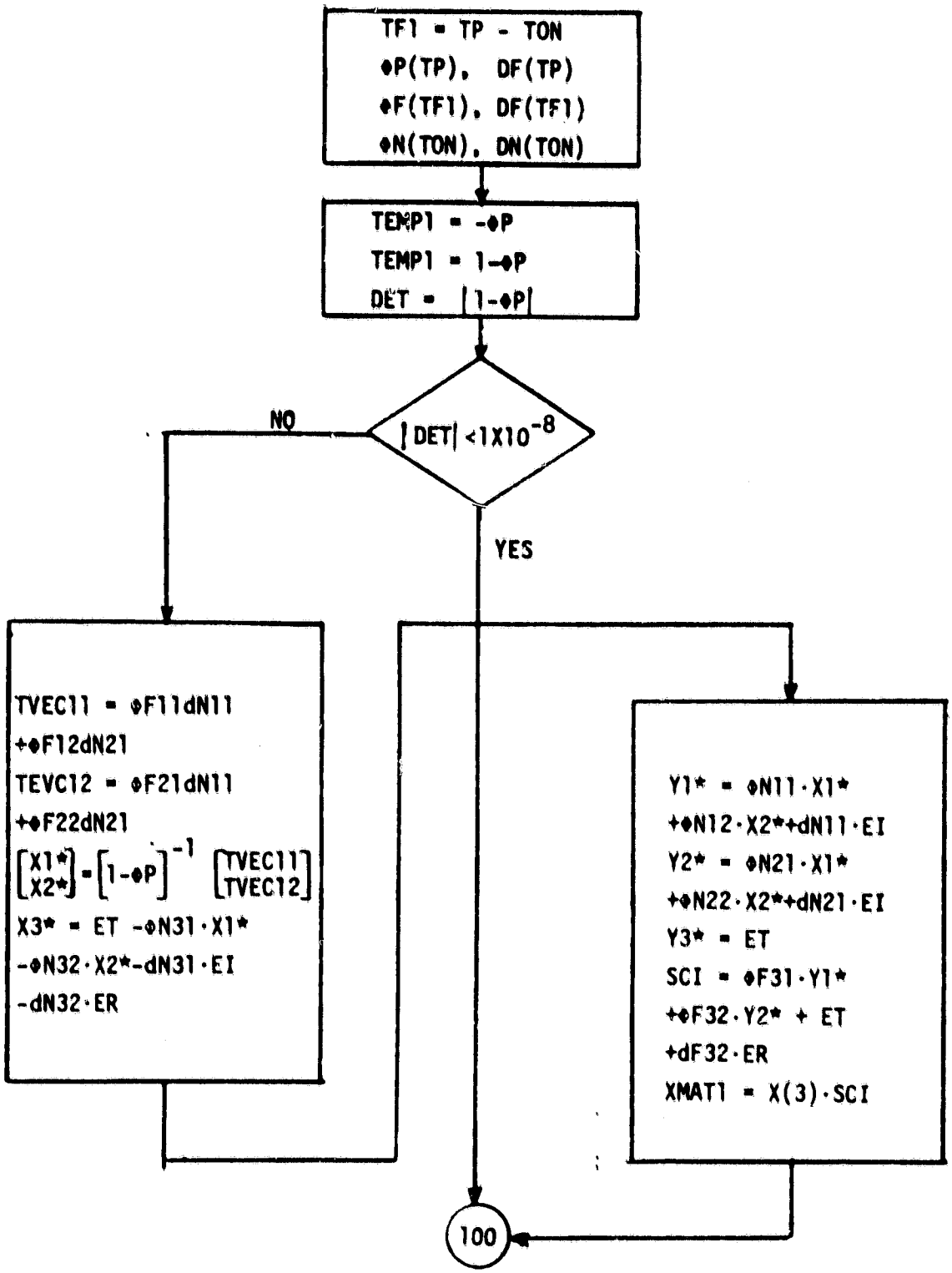


FIGURE A10 - PAS 1 SUBROUTINE XMAT 1

SUBROUTINE OVSH 1 (PSI, FP, N, K)

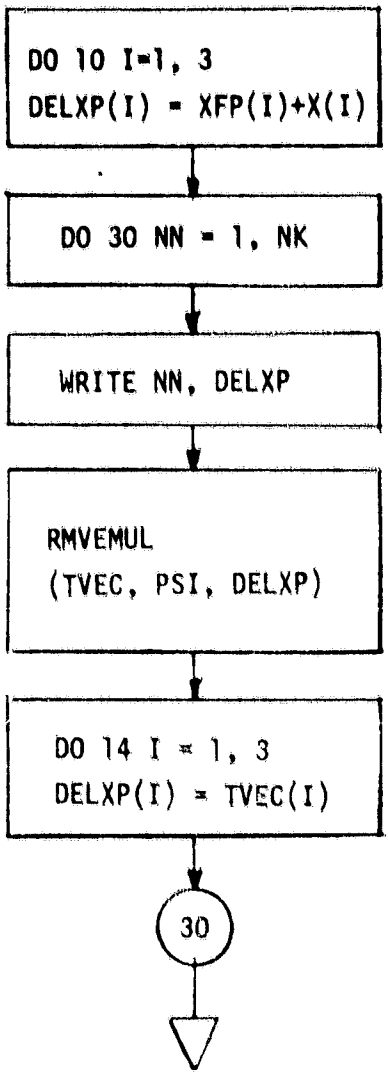


FIGURE A11 - PAS 1 SUBROUTINE OVSH 1

SUBROUTINE PHDMAT (PHI, D, T, F, G)

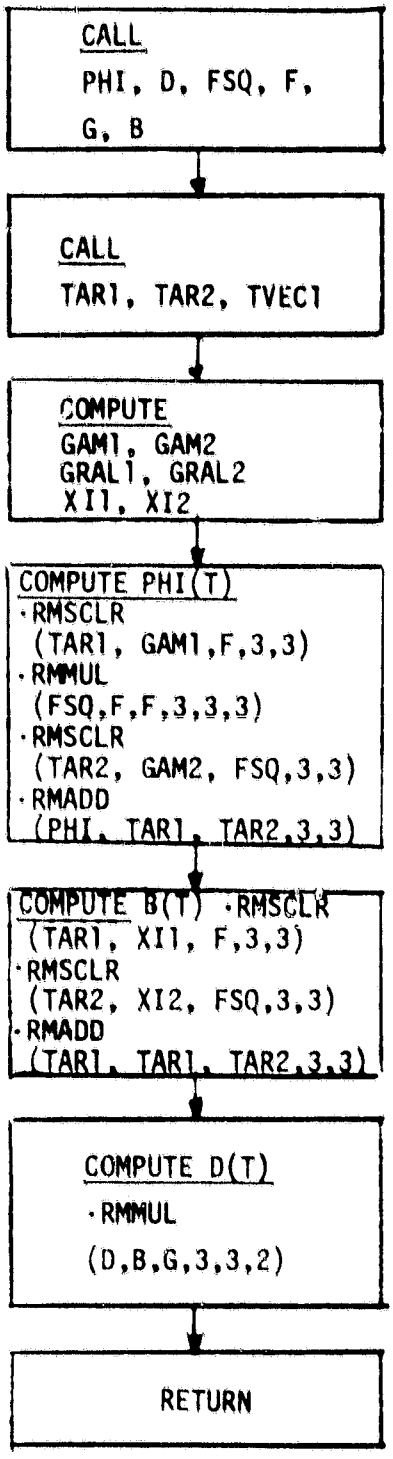
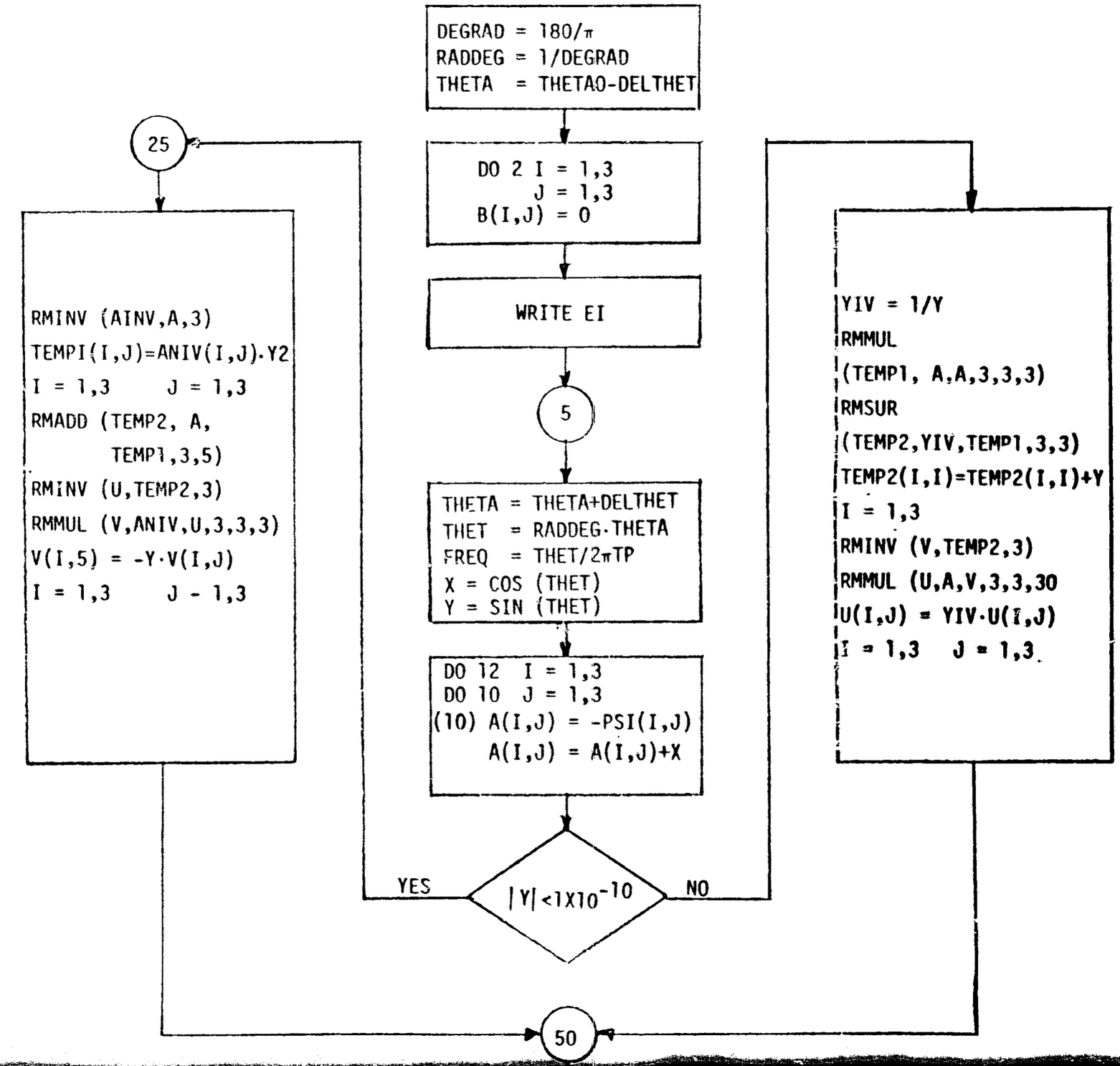


FIGURE A12 - PAS 1 SUBROUTINE PHDMAT

SUBROUTINE FREQ (PHI,DVEC,H,TP,THETA,THETAF, DELTHET, EI,ER)



FOLDOUT FRAME

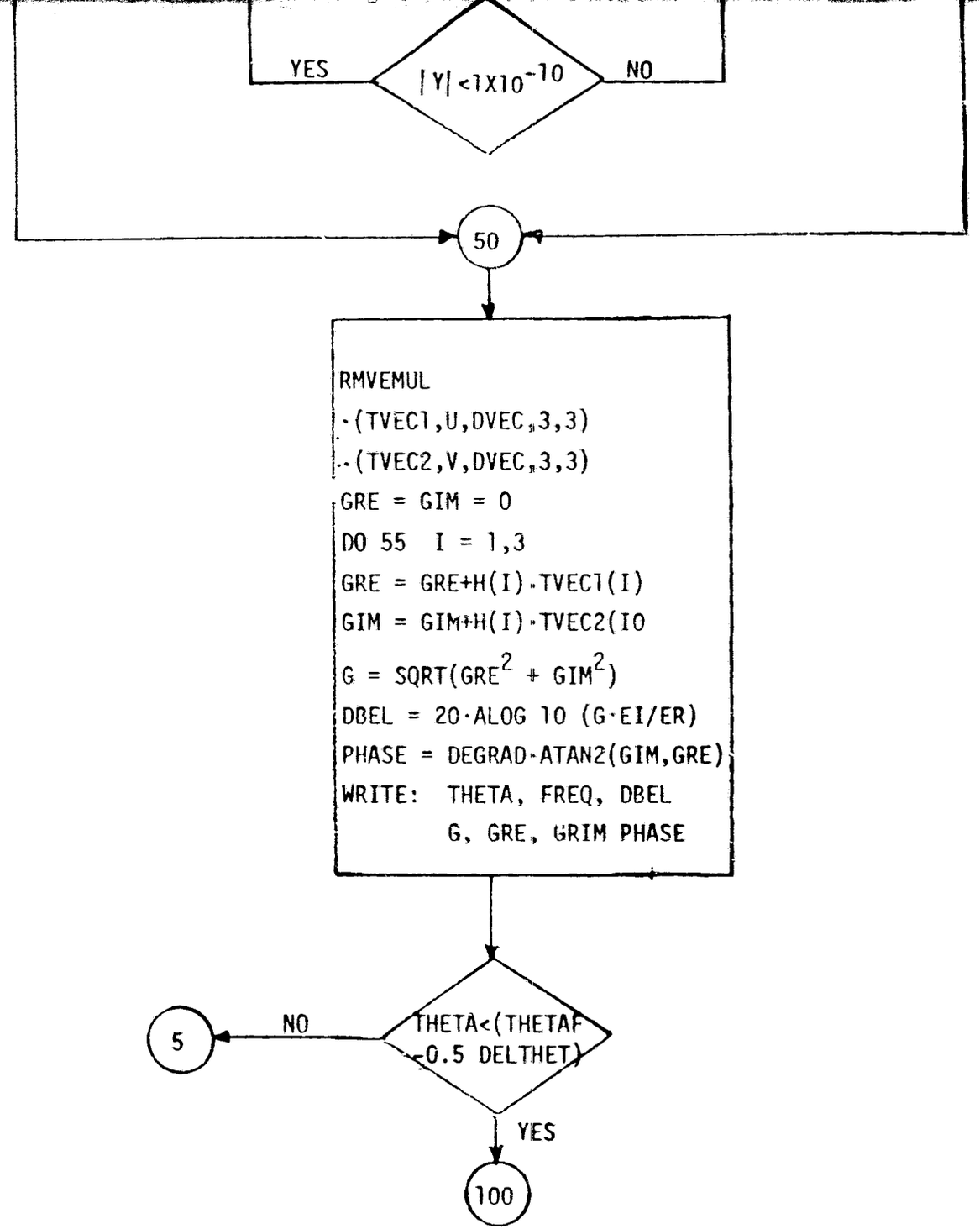


FIGURE A13 - SUBROUTINE FREQ

FOLDOUT FRAME 2

Figure A14. Generation of Closed-Loop Transfer Function for Subroutine FREQ

$$G(j\omega) = H(Ie^{j\omega T_p} - \Psi)^{-1} \Gamma, \quad 0 \leq \omega T_p \leq \pi$$

Let $\omega T_p = \theta$

$$(Ie^{j\theta} - \Psi)^{-1} = (jI\sin\theta + I\cos\theta - \psi)^{-1}$$

$$= [jI\sin\theta + A]^{-1} \qquad (\text{where } A = I\cos\theta - \psi)$$

Let $Q_1 + jQ_2 = [jI\sin\theta + A]^{-1}$

then

$$[A + jI\sin\theta][Q_1 + jQ_2] = I$$

$$AQ_1 - \sin\theta\, Q_2 = I$$

$$\sin\theta\, Q_1 + AQ_2 = 0$$

If $\sin\theta > 0 + \epsilon$, $Q_1 = -\dfrac{AQ_2}{\sin\theta}$

$$-\frac{A^2Q_2}{\sin\theta} - \sin\theta\, Q_2 = I$$

$$Q_2 = -\left[\frac{A^2}{\sin\theta} + I\sin\theta\right]^{-1}$$

$$Q_1 = \frac{A}{\sin\theta}\left[\frac{A^2}{\sin\theta} + I\sin\theta\right]^{-1}$$

If $\sin\theta \to 0$

$$Q_2 = -A^{-1}\sin\theta\, Q_1$$

$$AQ_1 + A^{-1}\sin^2\theta\, Q_1 = I$$

$$Q_1 = \left[A + A^{-1}\sin^2\theta\right]^{-1}$$

$$Q_2 = -A^{-1}\sin\theta\left[A + A^{-1}\sin^2\theta\right]^{-1}$$

APPENDIX A

BUCK PAS COMPUTER PROGRAM

```
00100       PROGRAM PAS1(INPUT,OUTPUT,TAPE5=INPUT,
00110      XTAPE6=OUTPUT,TAPE7)
00120       DIMENSION RIPX(3,1),PSI(3,3),PSY(3,3),GAM(3,1),INT(8),
00130      XXFP(3,1),PRAM(10),H(3),R(3,5),ITBL(3),IVD(3),DELX(3,1),
00140      XPHI1(3,3),PHI2(3,3),PHI3(3,3),D1(3,2),D2(3,2),D3(3,2),
00150      XTEMP1(3,3),YVEC1(3),PHIP(3,3),
00160      XRPRAM(10),XTR(3,1),YTR(3,1),ZTR(3,1)
00170       COMMON /PARAM/F1(3,3),F2(3,3),F3(3,3),G1(3,2),G2(3,2),
00180      XG3(3,2)
00190       COMMON/EXTPAR/NIT,EPS,TP,ET,MODE,TF,LDUTY
00200       COMMON /STATE/X(3,1),Y(3,1),Z(3,1),U(2,1)
00210       EQUIVALENCE (PRAM(1),C1),(PRAM(2),C2),(PRAM(3),R3),
00220      X (PRAM(4),R4),(PRAM(5),R5),(PRAM(6),RN2),(PRAM(7),XLO),
00230      X (PRAM(8),CO),(PRAM(9),RL),(PRAM(10),EI)
00240       REAL LP
00250       DATA EI, ER, ET, RL, TP/50.,20.,8.,10.,30.E-6/
00260       DATA EISWIT,XMU/60.,0.01/
00270       DATA XLO,RO,CO,R5,RN1,RN2/25.E-5,0.015,3.E-4,0.077,40.,26./
00280       DATA C1,C2,R1,R2,R3,R4/2200.E-12,0.022E-6,28.7E3,
00290      X13.5E3,10.E3,100.E3/
00300       DATA THETAO,DELTHET,THETAF,H/0.,5.,180.,1.,0.,0./
00310       DATA NIT,EPS/100,1.E-6/
00320       DATA MODE/2/
00330       DATA IPLOT,LIST,LPEAK,LFE,NK,LFREQ/0,0,0,0,15,0/
00340       DATA LRTL,NRL,DPRAM,PRAMF/0,2,0.,0./
00350       DATA LPARAM,LCOMP,LPC,LSA,LRLPC,LCFR,LPEAK/1,0,0,0,0,0,0/
00360       DATA RPRAM/2HC1,2HC2,2HR3,2HR4,2HR5,2HN2,2HLO,2HCO,
00370      X 2HRL,2HEI/
00380       DATA RLSWIT,LRL/600.,0./
00390       DATA TF,LDUTY,TPCON/8.5393E-6,2,3.E-5/
00400       DATA OLDTIME,LTR,LRESP,TSWIT,TFINAL/0.,1,0,2.E-3,13.9E-3/
00410       DATA NTR,SEPS/1,2.E-4/
00420       NAMELIST/PARAM/EI,ER,ET,RL,TP,XLO,RO,CO,R5,RN1,RN2,
00430      XC1,C2,R1,R2,R3,R4,EISWIT,TF
00440       NAMELIST/COMP/EPS,NIT,XMU
00450       NAMELIST/CONTRL/LPARAM,LCOMP,LPC,LSA,LRTL,LRLPC,LFREQ
00460      X,LCFR,LPEAK,LOAD,LDUTY,LRESP
00470       NAMELIST/RLPARAM/NRL,DPRAM,PRAMF
00480       NAMELIST/FRPARAM/THETAO,DELTHET,THETAF
00490       NAMELIST/TAPARAM/EI,EISWIT,NK
00500       NAMELIST/RPARAM/RL,RLSWIT,NK
00510       NAMELIST/TRPARAM/OLDTIME,TSWIT,TFINAL
00520       REWIND 7
00530 C     WRITE(6,99)
00540    99 FORMAT(1X,*PROGRAM FUNCTION*,13X,* CONTROL PARAMETER (1-YES 0-NO)*
00550      X /* CHANGE PARAM*,18X,*LPARAM*
00560      X /* CHANGE COMP*,19X,*LCOMP*
00570      X /* LIST PARAM, COMP*,14X,*LPC*
```

```
00580      X /* STABILITY ANALYSIS*,12X,*LSA*
00590      X /* ROOT LOCUS ANALYSIS*,11X,*LRTL*
00600      X /*      LIST PARAMETER CODE*,6X,*LRLPC*
00610      X /* AUDIO ANALYSIS*,16X,*LFREQ*
00620      X /*      CHANGE FREQ RANGE*,8X,*LCFR*
00630      X /* TRANSIENT ANALYSIS*,12X,*LPEAK*
00640      X /* TRANSIENT LOAD    *,12X,*LOAD*
00650      X /* DUTY CYCLE SCHEME *,12X,*LDUTY*
00660      X /* TRANSISTION ALGORITHM *,6X,*LRESP*/)
00670    400 CONTINUE
00680        WRITE(6,401)
00690    401 FORMAT(1X,*ENTER N TO DISCONTINUE PAS, OTHERWISE Y*)
00700        READ(5,402) X1
00710    402 FORMAT(A1)
00720        IF(X1.EQ.1HN) STOP
00730        WRITE(6,403)
00740    403 FORMAT(1X,*INPUT PAS CONTROL PARAMETERS*)
00750        READ(5,CONTRL)
00760 C      WRITE(6,CONTRL)
00770        IF(LDUTY.EQ.1) GO TO 420
00780        WRITE(6,421)
00790    421 FORMAT(/1X,*CONSTANT TF DUTY CYCLE SCHEME*/)
00800        GO TO 422
00810    420 CONTINUE
00820        TP=TPCON
00830        WRITE(6,423)
00840    423 FORMAT(/1X,*CONSTANT TP DUTY CYCLE SCHEME*/)
00850    422 CONTINUE
00860        IF(LPARAM.EQ.0) GO TO 404
00870        READ(5,PARAM)
00880    404 IF(LCOMP.EQ.0) GO TO 405
00890        READ(5,COMP)
00900    405 IF(LPC.EQ.0) GO TO 406
00910        WRITE(6,PARAM)
00920        WRITE(6,COMP)
00930    406 IF(LSA.EQ.0) GO TO 407
00940        GO TO 5
00950    407 IF(LRTL.EQ.0) GO TO 408
00960        IF(LRLPC.EQ.0) GO TO 409
00970        WRITE(6,460)
00980    460 FORMAT(1X,*  CODE    PARAMETER*
00990       X /*   1=     C1*/*   2=     C2*/*   3=     R3*/
01000       X *   4=     R4*/*   5=     R5*/*   6=     N2*/
01010       X *   7=     L0*/*   8=     C0*/*   9=     RL*/
01020       X *  10=     EI*/)
01030    409 WRITE(6,410)
01040    410 FORMAT(1X,*INPUT ROOT LOCUS PARAMETERS*)
01050        READ(5,RLPARAM)
01060        WRITE(6,RLPARAM)
01070        GO TO 5
01080    408 IF(LFREQ.EQ.0) GO TO 411
01090        IF(LCFR.EQ.0) GO TO 5
01100        WRITE(6,413)
01110    413 FORMAT(1X,*INPUT FREQUENCY RANGE PARAMETERS*)
```

```
01120       READ(5,FRPARAM)
01130       WRITE(6,FRPARAM)
01140       GO TO 5
01150   411 IF(LPEAK.EQ.0) GO TO 415
01160       WRITE(6,414)
01170   414 FORMAT(1X,*INPUT TRANSIENT ANALYSIS PARAMETERS*)
01180       READ(5,TAPARAM)
01190       WRITE(6,TAPARAM)
01200       GO TO 5
01210   415 IF(LOAD.EQ.0) GO TO 424
01220       WRITE(6,416)
01230   416 FORMAT(1X,*INPUT LOAD CHANGE PARAMETERS*)
01240       READ(5,RPARAM)
01250       WRITE(6,RPARAM)
01260   424 CONTINUE
01270       IF(LRESP.EQ.0) GO TO 400
01280       WRITE(6,425)
01290   425 FORMAT(1X,*INPUT TRANSITION RESPONSE PARAMETERS*)
01300       READ(5,TRPARAM)
01310       WRITE(6,TRPARAM)
01320     5 CONTINUE
01330       IF(LRTL.EQ.0)GO TO 4
01340       WRITE(6,212)RPRAM(NRL),PRAM(NRL)
01350   212 FORMAT(//*ROOT LOCUS PARAMETER *,A2,* = *,G12.4)
01360     4 CONTINUE
01370       OLDEI=EI
01380       OLDRL=RL
01390       RKD=R2/(R1+R2)
01400       RN=RN2/RN1
01410       DO 8 I=1,3
01420       G1(I,1)=G1(I,2)=G2(I,1)=G2(I,2)=G3(I,1)=G3(I,2)=0.
01430       DO 8 J=1,3
01440   8   F1(I,J)=F2(I,J)=F3(I,J)=0.
01450       F1(1,1)=F2(1,1)=-1./(RL*C0+R5*C0)-R5*RL/(XL0*RL+XL0*R5)
01460       F3(1,1)=-1./(C0*(R5+RL))
01470       F1(1,2)=F2(1,2)=F3(1,2)=RL/(C0*RL+C0*R5)-R0*R5*RL/(XL0*RL+
01480      XXL0*R5)
01490       F1(2,1)=F2(2,1)=-1./XL0
01500       F1(2,2)=F2(2,2)=F3(2,2)=-R0/XL0
01510       F1(3,1)=F2(3,1)=C2/(RL*C1*C0+R5*C1*C0)+RN/(R4*C1)-RKD/(R3*
01520      X C1)+C2*R5*RL/(C1*XL0*RL+C1*XL0*R5)
01530       F3(3,1)=-RKD/(R3*C1)+C2/(RL*C1*C0+R5*C1*C0)
01540       F1(3,2)=F2(3,2)=F3(3,2)=C2*R0*R5*RL/(C1*XL0*R5+C1*XL0*RL)-
01550      X RL*C2/(C1*C0*R5+C1*C0*RL)+RN*R0/(R4*C1)
01560       G1(1,1)=R5*RL/(XL0*R5+XL0*RL)
01570       G1(2,1)=1./XL0
01580       G1(3,1)=-RN/(R4*C1)-C2*R5*RL/(C1*XL0*R5+C1*XL0*RL)
01590       G1(3,2)=G2(3,2)=G3(3,2)=RKD/(R3*C1)
01600       U(1,1)=EI
01610       U(2,1)=ER
01620       P0=ER**2/RL
01630       LP=XL0*P0
01640     7 CONTINUE
01650       IF(LDUTY.EQ.1) GO TO 41
```

```
01660       DN=EI*(EI-ER)
01670       TON=(LP+SQRT(LP**2+2.*DN*LP*TF))/DN
01680       TF1=TON*(EI-ER)/ER
01690       TF2=TF-TF1
01700       TP=TON+TF
01710       GO TO 42
01720    41 CONTINUE
01730       GTON=SQRT(2.*LP*TP)
01740       TON=GTON/SQRT(EI*(EI-ER))
01750       TF1=TON*(EI-ER)/ER
01760       TF2=TP-TON-TF1
01770    42 CONTINUE
01780       TEPS=EPS
01790       DELTON=XMU*TON
01800       IF(TF2.GE.TEPS) GO TO 6
01810       MODE=1
01820       IF(LDUTY.EQ.1) GO TO 43
01830       TON=TF*ER/(EI-ER)
01840       TF1=TF
01850       TP=TON+TF1
01860       GO TO 44
01870    43 TON=TP*ER/EI
01880       TF1=TP-TON
01890    44 CONTINUE
01900       OTON=TON
01910       OTF1=TF1
01920       DELTON=XMU*TON
01930       IT=0
01940       SC1=XMAT1(TON,EI,ER)
01950    19 DMATCH=(XMAT1(TON+DELTON,EI,ER)-SC1)/DELTON
01960       TON=TON-SC1/DMATCH
01970       SC1=XMAT1(TON,EI,ER)
01980       IT=IT+1
01990       IF(ABS(SC1).LE.EPS) GO TO 20
02000       IF(IT.LT.NIT) GO TO 19
02010       TON=OTON
02020    20 CONTINUE
02030       IF(LDUTY.EQ.1) GO TO 45
02040       TF1=TF
02050       TP=TON+TF1
02060       GO TO 46
02070    45 CONTINUE
02080       TF1=TP-TON
02090    46 CONTINUE
02100       CALL PHDMAT(PHIP,D2,TP,F1,G1)
02110       CALL PHDMAT(PHI2,D2,TF1,F1,G1)
02120       CALL PHDMAT(PHI1,D1,TON,F1,G1)
02130       DO 22 I=1,2
02140       DO 21 J=1,2
02150    21 TEMP1(I,J)=-PHIP(I,J)
02160    22 TEMP1(I,I)=1.+TEMP1(I,I)
02170       DET=TEMP1(1,1)*TEMP1(2,2)-TEMP1(2,1)*TEMP1(1,2)
02180       IF(ABS(DET).LT.1.E-8) GO TO 250
02190       TVEC1(1)=PHI2(1,1)*D1(1,1)+PHI2(1,2)*D1(2,1)
```

```
02200       TVEC1(2)=PHI2(2,1)*D1(1,1)+PHI2(2,2)*D1(2,1)
02210       X(1,1)=EI*(TEMP1(2,2)*TVEC1(1)-TEMP1(1,2)*TVEC1(2))/DET
02220       X(2,1)=EI*(TEMP1(1,1)*TVEC1(2)-TEMP1(2,1)*TVEC1(1))/DET
02230       X(3,1)=ET-PHI1(3,1)*X(1,1)-PHI1(3,2)*X(2,1)-D1(3,1)*EI
02240      X -D1(3,2)*ER
02250       Y(1,1)=PHI1(1,1)*X(1,1)+PHI1(1,2)*X(2,1)+D1(1,1)*EI
02260       Y(2,1)=PHI1(2,1)*X(1,1)+PHI1(2,2)*X(2,1)+D1(2,1)*EI
02270       Y(3,1)=ET
02280       SC1=PHI2(3,1)*Y(1,1)+PHI2(3,2)*Y(2,1)+ET+D2(3,2)*ER
02290       DO 32 I=1,3
02300    32 RIPX(I,1)=X(I,1)-Y(I,1)
02310 C     IF(LRTL.EQ.2) GO TO 36
02320       TPCT=100.*TON/TP
02330       WRITE(6,35) EI,RL,MODE,TPCT,TON,TF1,TP,X,Y,RIPX,SC1,IT,TEPS
02340    35 FORMAT(/*EI=*,G15.4,*  RL=*,G15.4,*  MODE=*,I3,
02350      X *  TPCT=*,G15.4/*TON=*,G15.4,*  TF1=*,G15.4,*  TP=*,
02360      X G15.4/*    X=*,3G15.4/*    Y=*,3G15.4/*RIPX=*,3G15.4/
02370      X * SC1=*,G15.4,*  IT=*,I3,*  TEPS=*,G15.4)
02380    36 CONTINUE
02390       GO TO 56
02400     6 CONTINUE
02410       MODE=2
02420       PTON=TON
02430       PTF1=TF1
02440       CALL STATE1(TON,TF1)
02450 C     PRINT 17,TON,TF1,TF2,X,Y,Z
02460 C  17 FORMAT(*APPROXIMATE STEADY STATE*/*TON=*,G15.6,
02470 C    X*  TF1=*,G15.6,*  TF2=*,G15.6/
02480 C    X*X=*,3G15.6/*Y=*,3G15.6/*Z=*,3G15.6//)
02490       IT=0
02500       CALL SMAT1(TON,TF1,TF2,SC1,XMU,XLO,PO)
02510     9 CALL SMAT1(TON+DELTON,TF1,TF2,SC2,XMU,XLO,PO)
02520       DSMAT1=(SC2-SC1)/DELTON
02530       TON=TON-SC1/DSMAT1
02540       CALL SMAT1(TON,TF1,TF2,SC1,XMU,XLO,PO)
02550       IT=IT+1
02560       IF(ABS(SC1).LE.EPS) GO TO 10
02570       IF(IT.LT.NIT) GO TO 9
02580       TF1=PTF1
02590       TON=PTON
02600       CALL SMAT1(TON,TF1,TF2,SC1,XMU,XLO,PO)
02610    10 CONTINUE
02620       DO 12 I=1,3
02630    12 RIPX(I,1)=X(I,1)-Y(I,1)
02640 C     IF(LRTL.EQ.2) GO TO 56
02650       TPCT=100.*TON/TP
02660 C     IF(LRTL.EQ.2) GO TO 56
02670       IF(MTR.EQ.2) GO TO 36
02680       WRITE(6,55) EI,RL,MODE,TPCT,TON,TF1,TF2,TP,X,Y,Z,RIPX,SC1,IT,TEPS
02690    55 FORMAT(/*EI=*,G15.4,*  RL=*,G15.4,*  MODE=*,I3,*  TPCT=*,
02700      X G15.4/*TON=*,G12.4,*  TF1=*,G12.4,*  TF2=*,G12.4,*  TP=*,
02710      X G12.4/*    X=*,3G15.4/*    Y=*,3G15.4/*    Z=*,3G15.4/
02720      X *RIPX=*,3G15.4/* SC1=*,G15.4,*  IT=*,I3,*  TEPS=*,G15.4)
02730    57 CONTINUE
```

```
02740     54 CONTINUE
02750        CALL PSIMAT(PSI,TON,TF1,X,U,XMU)
02760        CALL RMCPY(PSY,PSI,3,3)
02770        ITBL(1)=3
02780        ITBL(3)=0
02790        CALL QRAL(PSY,R,3,M,V,INT,IVD,ITBL)
02800        IF(LRTL.NE.2) GO TO 67
02810 C      WRITE(6,68) MODE,TPCT
02820     67 CONTINUE
02830     68 FORMAT(*MODE =*,I2,* DUTY CYCLE =*,F5.2/)
02840        WRITE(6,70) ((PSI(I,J),J=1,3),I=1,3)
02850     70 FORMAT(/*PSI=*,3G15.4/2(4X,3G15.4/))
02860        IF(MTR.EQ.2) GO TO 76
02870     72 WRITE(6,74)((R(I,J),J=1,2),I=1,3)
02880     74 FORMAT(/4X,*REAL*,11X,*IMAG*,7X,2G15.4/4G15.4)
02890     76 CONTINUE
02900        IF(LRESP.EQ.0) GO TO 550
02910        CALL PHDMAT(PHI1,D1,TON,F1,G1)
02920        CALL PHDMAT(PHI2,D2,TF1,F2,G2)
02930        IF(MODE.EQ.1) GO TO 589
02940        CALL PHDMAT(PHI3,D3,TF2,F3,G3)
02950    589 CONTINUE
02960        TIME=OLDTIME
02970        IF(LTR.EQ.2) GO TO 501
02980        IF(MODE.EQ.1) GO TO 500
02990        CALL RMCPY(ZTR,Z,3,1)
03000        VOTR=ZTR(1,1)-ER
03010 C      WRITE(6,599)TIME,ZTR,VOTR
03020        WRITE(7,599)TIME,ZTR,VOTR
03030    599 FORMAT(6G12.4)
03040        GO TO 501
03050    500 CONTINUE
03060        CALL RMCPY(YTR,Y,3,1)
03070        VOTR=YTR(1,1)-ER
03080 C      WRITE(6,599)TIME,YTR,VOTR
03090        WRITE(7,599)TIME,YTR,VOTR
03100    501 CONTINUE
03110        TIME=TIME+TON
03120        IF(MODE.EQ.1) GO TO 502
03130        CALL STS1(XTR,PHI3,ZTR,D3,U)
03140        GO TO 503
03150    502 CONTINUE
03160        CALL STS1(XTR,PHI2,YTR,D2,U)
03170    503 CONTINUE
03180        VOTR=XTR(1,1)-ER
03190 C      WRITE(6,599)TIME,XTR,VOTR
03200        WRITE(7,599)TIME,XTR,VOTR
03210        TIME=TIME+TF1
03220        CALL STS1(YTR,PHI1,XTR,D1,U)
03230        VOTR=YTR(1,1)-ER
03240 C      WRITE(6,599)TIME,YTR,VOTR
03250        WRITE(7,599)TIME,YTR,VOTR
03260        IF(MODE.EQ.1) GO TO 504
03270        TIME=TIME+TF2
```

```
03280       CALL STS1(ZTR,PHI2,YTR,D2,U)
03290       VOTR=ZTR(1,1)-ER
03300 C     WRITE(6,599)TIME,ZTR,VOTR
03310       WRITE(7,599)TIME,ZTR,VOTR
03320  504  CONTINUE
03330       IF(TIME.GT.TFINAL) GO TO 505
03340       IF(TIME.LT.TSWIT) GO TO 501
03350       IF(LTR.EQ.2) GO TO 501
03360       IF(MODE.EQ.2) GO TO 506
03370       CALL RMCPY(ZTR,YTR,3,1)
03380  506  CONTINUE
03390       U(1,1)=EISWIT
03400       EI=EISWIT
03410       OLDTIME=TIME
03420       LTR=2
03430       GO TO 7
03440  505  CONTINUE
03450       OLDTIME=0.
03460       LTR=1
03470       MTR=2
03480       LRESP=0
03490       GO TO 7
03500  550  CONTINUE
03510       MTR=1
03520       IF(LFREQ.EQ.0) GO TO 150
03530       CALL GAMM1(GAM,TON,TF1,X,U,XMU)
03540       WRITE(6,75)  (GAM(I,1),I=1,3)
03550    75 FORMAT(/*GAM=*/3(G15.6/))
03560    69 CALL FREQ1(PSI,GAM,H,THETA0,THETAF,DELTHET,EI,ER)
03570       READ(5,FRPARAM)
03580       IF(THETAF.EQ.0.)GO TO 149
03590       WRITE(6,FRPARAM)
03600       GO TO 69
03610   149 CONTINUE
03620       LFREQ=0
03630  150  CONTINUE
03640       IF(LFE.EQ.1) GO TO 160
03650       IF(LPEAK.EQ.0) GO TO 200
03660       DO 152 I=1,3
03670  152  XFP(I,1)=X(I,1)
03680       EI=EISWIT
03690       U(1,1)=EI
03700       LFE=1
03710       WRITE(6,189)
03720  189  FORMAT(1X,*SET UP EI STEP INPUT*)
03730       GO TO 7
03740  160  CALL OVSH1(PSI,XFP,NK)
03750  200  CONTINUE
03760       LPEAK=LFE=0
03770       EI=OLDEI
03780       U(1,1)=EI
03790       IF(LRL.EQ.1) GO TO 161
03800       IF(LOAD.EQ.0) GO TO 201
03810       RL=RLSWIT
```

```
03820       LRL=1
03830       DO 153 I=1,4
03840  153  XFP(I,1)=X(I,1)
03850       WRITE(6,188)
03860  188  FORMAT(1X,*SET UP RL LOAD CHANGE*)
03870       GO TO 4
03880  161  CALL OVSH1(PSI,XFP,NK)
03890  201  CONTINUE
03900       LOAD=LRL=0
03910       RL=OLDRL
03920       IF(LRTL.EQ.0) GO TO 300
03930       PRAM(NRL)=PRAM(NRL)+DPRAM
03940       IF(PRAM(NRL).GT.PRAMF) GO TO 300
03950       LRTL=2
03960       GO TO 5
03970   250 WRITE(6,252)IT,X
03980   252 FORMAT(*UNSCHEDULED TERMINATION IT=*,I3,/*X=*,3G15.6/)
03990  300  CONTINUE
04000       GO TO 400
04010       END
04020 C
04030 C
04040       SUBROUTINE OVSH1(PSI,XFP,NK)
04050       DIMENSION PSI(3,3),XFP(3,1),DELXP(3,1),TVEC(3,1)
04060       COMMON/STATE/X(3,1),Y(3,1),Z(3,1),U(2,1)
04070       DO 10 I=1,3
04080   10  DELXP(I,1)=XFP(I,1)-X(I,1)
04090       DO 30 NN=1,NK
04100       WRITE(6,12) NN,DELXP
04110   12  FORMAT(I3,3G15.6)
04120       CALL RMMUL(TVEC,PSI,DELXP,3,3,1)
04130       DO 14 I=1,3
04140   14  DELXP(I,1)=TVEC(I,1)
04150   30  CONTINUE
04160       RETURN
04170       END
04180 C
04190 C
04200       FUNCTION ZETA1(T,X,U)
04210       DIMENSION PHI1(3,3),D1(3,2),X(3,1),U(2,1)
04220       COMMON/PARAM/F1(3,3),F2(3,3),F3(3,3),G1(3,2),
04230      .XG2(3,2),G3(3,2)
04240       COMMON/EXTPAR/NIT,EPS,TP,ET,MODE,TF,LDUTY
04250       CALL PHDMAT(PHI1,D1,T,F1,G1)
04260       ZETA1=-ET+PHI1(3,1)*X(1,1)+PHI1(3,2)*X(2,1)+PHI1(3,3)*X(3,1)+
04270      X D1(3,1)*U(1,1)+D1(3,2)*U(2,1)
04280       RETURN
04290       END
04300 C
04310 C
04320       SUBROUTINE FREQ1(PSI,DVEC,H,THETA0,THETAF,DELTHET,EI,ER)
04330       DIMENSION PSI(3,3),DVEC(3,1),H(3)
04340       DIMENSION A(3,3),AINV(3,3),B(3,3),U(3,3),V(3,3),TEMP1(3,3)
04350       DIMENSION TEMP2(3,3),TVEC1(3,1),TVEC2(3,1)
```

```
04360       COMMON/EXTPAR/NIT,EPS,TP,ET,MODE,TF,LDUTY
04370       DEGRAD=180./3.1415927
04380       RADDEG=1./DEGRAD
04390       THETA=THETAO-DELTHET
04400       DO 2 I=1,3
04410       DO 2 J=1,3
04420     2 B(I,J)=0.
04430       WRITE(6,1) EI
04440     1 FORMAT(//*EI=*,G12.6/,
04450      X * THETA   FREQ (HZ)   DBEL*,5X,*G*,11X,*REG*,10X,*IMG*,
04460      X 6X,*PHASE*)
04470     5 CONTINUE
04480       THETA=THETA+DELTHET
04490       THET=RADDEG*THETA
04500       FRE=THET/(6.2831853*TP)
04510       RX=COS(THET)
04520       RY=SIN(THET)
04530       DO 12 I=1,3
04540       DO 10 J=1,3
04550    10 A(I,J)=-PSI(I,J)
04560       A(I,I)=A(I,I)+RX
04570    12 B(I,I)=RY
04580       IF(ABS(RY).LT.1.E-10) GO TO 25
04590       RYIV=1./RY
04600       CALL RMMUL(TEMP1,A,A,3,3,3)
04610       CALL RMSCLR(TEMP2,RYIV,TEMP1,3,3)
04620       DO 14 I=1,3
04630    14 TEMP2(I,I)=TEMP2(I,I)+RY
04640       CALL RMINV(V,TEMP2,3)
04650       CALL RMMUL(U,A,V,3,3,3)
04660       DO 16 I=1,3
04690       DO 16 J=1,3
04700       U(I,J)=RYIV*U(I,J)
04710    16 V(I,J)=-V(I,J)
04720       GO TO 50
04730    25 CALL RMINV(AINV,A,3)
04740       DO 30 I=1,3
04750       DO 30 J=1,3
04760    30 TEMP1(I,J)=AINV(I,J)*RY**2
04770       CALL RMADD(TEMP2,A,TEMP1,3,3)
04780       CALL RMINV(U,TEMP2,3)
04790       CALL RMMUL(V,AINV,U,3,3,3)
04800       DO 34 I=1,3
04810       DO 34 J=1,3
04820    34 V(I,J)=-RY*V(I,J)
04830    50 CONTINUE
04840       CALL RMMUL(TVEC1,U,DVEC,3,3,1)
04850       CALL RMMUL(TVEC2,V,DVEC,3,3,1)
04860       GRE=GIM=0.
04870       DO 55 I=1,3
04880       GRE=GRE+H(I)*TVEC1(I,1)
04890    55 GIM=GIM+H(I)*TVEC2(I,1)
04900       G=SQRT(GRE**2+GIM**2)
04910       DBEL=20.*ALOG10(G*EI/ER)
```

```
04920       PHASE=DEGRAD*ATAN2(GIM,GRE)
04930       WRITE(6,60) THETA,FRE,DBEL,G,GRE,GIM,PHASE
04940    60 FORMAT(F6.2,G12.4,F9.2,3G12.4,F9.2)
04670       WRITE(7,59)FRE,DBEL,PHASE
04680    59 FORMAT(3G15.6)
04950       IF(THETA.LT.THETAF-0.5*DELTHET) GO TO 5
04960   100 CONTINUE
04970       RETURN
04980       END
04990 C
05000 C
05010       SUBROUTINE PHDMAT(PHI,D,T,F,G)
05020       DIMENSION PHI(3,3),D(3,2),FSQ(3,3),F(3,3),S(3,2),B(3,3)
05030       DIMENSION TAR1(3,3), TAR2(3,3), TVEC1(3)
05040       ALPHA=-0.5*(F(1,1)+F(2,2))
05050       DET=0.25*(F(1,1)+F(2,2))**2-F(1,1)*F(2,2)+F(1,2)*F(2,1)
05060       IF(DET.LT.0.) GO TO 20
05070       A=-ALPHA+SQRT(DET)
05080       B=-ALPHA-SQRT(DET)
05090       EXPA=EXP(A*T)-1.
05100       EXNA=EXP(-A*T)-1.
05110       EXPB=EXP(B*T)-1.
05120       EXNB=EXP(-B*T)-1.
05130       CC1=A*B*(A-B)
05140       GAM1=(A**2*EXPB-B**2*EXPA)/CC1
05150       GAM2=(B*EXPA-A*EXPB)/CC1
05160       CC2=A**2*(A-B)
05170       CC3=B**2*(A-B)
05180       XI1=-(A+B)*T/(A*B)-A*EXNB/CC3+B*EXNA/CC2
05190       XI2=T/(A*B)-EXNA/CC2+EXNB/CC3
05200       GO TO 30
05210    20 BETA=SQRT(-DET)
05220       CC3=1./(ALPHA**2+BETA**2)
05230       CC1=2.*ALPHA*CC3
05240       CC2=(ALPHA**2-BETA**2)/(2.*ALPHA*BETA)
05250       CC4=ALPHA/BETA
05260       SB=SIN(BETA*T)
05270       CB=COS(BETA*T)
05280       EMAL=EXP(-ALPHA*T)
05290       EPAL=EXP(ALPHA*T)
05300       GAM1=CC1*(1.-EMAL*(CC2*SB+CB))
05310       GAM2=CC3*(1.-EMAL*(CC4*SB+CB))
05320       GRAL1=CC3*(EPAL*(ALPHA*CB+BETA*SB)-ALPHA)
05330       GRAL2=CC3*(EPAL*(ALPHA*SB-BETA*CB)+BETA)
05340       XI1=CC1*(T-GRAL1+CC2*GRAL2)
05350       XI2=CC3*(T-GRAL1+CC4*GRAL2)
05360    30 CALL RMSCLR(TAR1,GAM1,F,3,3)
05370       CALL RMMUL(FSQ,F,F,3,3,3)
05380       CALL RMSCLR(TAR2,GAM2,FSQ,3,3)
05390       CALL RMADD(PHI,TAR1,TAR2,3,3)
05400       DO 10 I=1,3
05410    10 PHI(I,I)=PHI(I,I)+1.
05420 C   MATRIX PHI(T) HAS BEEN COMPUTED
05430       CALL RMSCLR(TAR1,XI1,F,3,3)
```

```
05440       CALL RMSCLR(TAR2,XI2,FSQ,3,3)
05450       CALL RMADD(TAR1,TAR1,TAR2,3,3)
05460       DO 12 I=1,3
05470    12 TAR1(I,I)=TAR1(I,I)+T
05480       CALL RMMUL(B,PHI,TAR1,3,3,3)
05490 C   MATRIX B(T) HAS BEEN COMPUTED
05500       CALL RMMUL(D,B,G,3,3,2)
05510 C MATRIX D(T) HAS BEEN COMPUTED
05520       RETURN
05530       END
05540 C
05550 C
05560       SUBROUTINE STATE1(TON,TF1)
05570       DIMENSION PHI1(3,3),PHI2(3,3),PHI3(3,3),D1(3,2),D2(3,2),
05580      XD3(3,2)
05590       COMMON/PARAM/ F1(3,3),F2(3,3),F3(3,3),G1(3,2),G2(3,2),
05600      XG3(3,2)
05610       COMMON/EXTPAR/NIT,EPS,TP,ET,MODE,TF,LDUTY
05620       COMMON/STATE/X(3,1),Y(3,1),Z(3,1),U(2,1)
05630       TF2=TP-TON-TF1
05640       IF(LDUTY.EQ.2)TF2=TF-TF1
05650       CALL PHDMAT(PHI1,D1,TON,F1,G1)
05660       CALL PHDMAT(PHI2,D2,TF1,F2,G2)
05670       CALL PHDMAT(PHI3,D3,TF2,F3,G3)
05680       A=1.-PHI3(1,1)*(PHI2(1,1)*PHI1(1,1)+PHI2(1,2)*PHI1(2,1))
05690      X -PHI3(1,2)*(PHI2(2,1)*PHI1(1,1)+PHI2(2,2)*PHI1(2,1))
05700       B=PHI3(1,1)*(PHI2(1,1)*D1(1,1)+PHI2(1,2)*D1(2,1)+D2(1,1))
05710       C=PHI3(1,2)*(PHI2(2,1)*D1(1,1)+PHI2(2,2)*D1(2,1)+D2(2,1))
05720       IF(ABS(A).LT.EPS) GO TO 250
05730       X(1,1)=(B+C+D3(1,1))*U(1,1)/A
05740       X(2,1)=0.
05750       X(3,1)=ET-PHI1(3,1)*X(1,1)-PHI1(3,2)*X(2,1)-
05760      X D1(3,1)*U(1,1)-D1(3,2)*U(2,1)
05770       CALL STS1(Y,PHI1,X,D1,U)
05780       CALL STS1(Z,PHI2,Y,D2,U)
05790       GO TO 23
05800   250 WRITE(6,252) A
05810   252 FORMAT(/*UNSCHEDULED TERMINATION*/*A=*,G12.6)
05820    23 RETURN
05830       END
05840 C
05850 C
05860       SUBROUTINE STS1(W2,PHI,W1,D,U)
05870       DIMENSION PHI(3,3),W1(3,1),W2(3,1),D(3,2),U(2,1),
05880      XTEMPY1(3,1),TEMPY2(3,1)
05890       CALL RMMUL(TEMPY1,PHI,W1,3,3,1)
05900       CALL RMMUL(TEMPY2,D,U,3,2,1)
05910       CALL RMADD(W2,TEMPY1,TEMPY2,3,1)
05920       RETURN
05930       END
05940 C
05950 C
05960       SUBROUTINE PSIMAT(PSI,TON,TF1,X,U,XMU)
05970       DIMENSION X(3,1),FBAR0(3,1),FBAR(3,1),PSI(3,3),
```

```
05980       XU(2,1),Y(3,1),PHI1(3,3),D1(3,3),
05990       XTEMPY1(3,1),TEMPY2(3,1),DELTX(3,1)
06000        COMMON/PARAM/F1(3,3),F2(3,3),F3(3,3),G1(3,2),
06010       XG2(3,2),G3(3,2)
06020        COMMON/EXTPAR/NIT,EPS,TP,ET,MODE,TF,LDUTY
06030        VARTON=TON
06040        VARTF1=TF1
06050        DELTON=XMU*TON
06060        DELTF1=XMU*TF1
06070        DO 71 I=1,3
06080     71 DELTX(I,1)=XMU*ABS(X(I,1))
06090        DELTX(2,1)=XMU
06100        CALL FFUNC1(VARTON,VARTF1,X,U,FBAR0)
06110 C      PRINT 51,FBAR0
06120     51 FORMAT(*FBAR0=*,G15.6/2(6X,G15.6/))
06130        DO 68 J=1,3
06140 C      IF(J.NE.2) GO TO 54
06150 C      IF(MODE.EQ.2) GO TO 72
06160     54 X(J,1)=X(J,1)+DELTX(J,1)
06170        IT=0
06180        SC1=ZETA1(VARTON,X,U)
06190     67 DZETA1=(ZETA1(VARTON+DELTON,X,U)-SC1)/DELTON
06200        IT=IT+1
06210        VARTON=VARTON-SC1/DZETA1
06220        SC1=ZETA1(VARTON,X,U)
06230        IF(ABS(SC1).LT.EPS) GO TO 64
06240        IF(IT.LT.NIT) GO TO 67
06250        WRITE(6,61) IT,SC1
06260     61 FORMAT(*MAX ITERATION ON TON. IT=*,I3,* SC1=*,G12.6/)
06270        GO TO 70
06280     64 IF(MODE.EQ.1) GO TO 65
06290        CALL PHDMAT(PHI1,D1,VARTON,F1,G1)
06300        CALL STS1(Y,PHI1,X,D1,U)
06310        IT=0
06320        B1=BMAT1(VARTF1,Y,U)
06330     63 DB=(BMAT1(VARTF1+DELTF1,Y,U)-B1)/DELTF1
06340        IT=IT+1
06350        VARTF1=VARTF1-B1/DB
06360        B1=BMAT1(VARTF1,Y,U)
06370        IF(ABS(B1).LT.EPS) GO TO 65
06380        IF(IT.LT.NIT) GO TO 63
06390        WRITE(6,66) IT,B1
06400     66 FORMAT(*MAX ITERATION ON TF1. IT=*,I3,* SC1=*,G12.6/)
06410        GO TO 70
06420     65 CALL FFUNC1(VARTON,VARTF1,X,U,FBAR)
06430 C      PRINT 53,VARTON,VARTF1
06440     53 FORMAT(*VARTON=*,G15.6/*VARTF1=*,G15.6/)
06450        DO 69 I=1,3
06460     69 PSI(I,J)=(FBAR(I,1)-FBAR0(I,1))/DELTX(J,1)
06470 C      PRINT 52,FBAR
06480     52 FORMAT(*FBAR=*,G15.6/3(5X,G15.6/))
06490        X(J,1)=X(J,1)-DELTX(J,1)
06500        GO TO 68
06510     72 DO 74 I=1,3
```

```
06520    74 PSI(I,2)=0.
06530    68 CONTINUE
06540       IF(MODE.EQ.2) GO TO 70
06550       CALL PHDMAT(PHI1,D1,TP,F1,G1)
06560       CALL RMADD(PSI,PSI,PHI1,3,3)
06570    70 RETURN
06580       END
06590 C
06600 C
06610       FUNCTION BMAT1(TF1,Y,U)
06620       DIMENSION PHI2(3,3),D2(3,2),Y(3,1),U(2,1)
06630       COMMON/PARAM/ F1(3,3),F2(3,3),F3(3,3),G1(3,2),G2(3,2),
06640      XG3(3,2)
06650       COMMON/EXTPAR/NIT,EPS,TP,ET,MODE,TF,LDUTY
06660       CALL PHDMAT(PHI2,D2,TF1,F2,G2)
06670       BMAT1=PHI2(2,1)*Y(1,1)+PHI2(2,2)*Y(2,1)
06680       RETURN
06690       END
06700 C
06710 C
06720       SUBROUTINE FFUNC1(TON,TF1,X,U,F)
06730       DIMENSION TEMP1(3,3),TEMP2(3,3),PHI1(3,3),PHI2(3,3),
06740      XPHI3(3,3),D1(3,2),D2(3,2),D3(3,2),PHI(3,3),V(3,3),
06750      XFTEMP1(3,1),FTEMP2(3,1),F(3,1),X(3,1),U(2,1),TVEC1(3,1)
06760       COMMON/PARAM/F1(3,3),F2(3,3),F3(3,3),G1(3,2),G2(3,2),
06770      XG3(3,2)
06780       COMMON/EXTPAR/NIT,EPS,TP,ET,MODE,TF,LDUTY
06790       IF(MODE.EQ.2) GO TO 1
06800       TF1=TP-TON
06810       IF(LDUTY.EQ.2)TF1=TF
06820       CALL PHDMAT(PHI1,D1,TON,F1,G1)
06830       CALL PHDMAT(PHI2,D2,TF1,F1,G1)
06840       TVEC1(1,1)=D1(1,1)*U(1,1)
06850       TVEC1(2,1)=D1(2,1)*U(1,1)
06860       TVEC1(3,1)=D1(3,1)*U(1,1)+D1(3,2)*U(2,1)
06870       CALL RMMUL(F,PHI2,TVEC1,3,3,1)
06880       F(3,1)=F(3,1)+D2(3,2)*U(2,1)
06890       GO TO 2
06900     1 TF2=TP-TF1-TON
06910       IF(LDUTY.EQ.2)TF2=TF-TF1
06920       CALL PHDMAT(PHI1,D1,TON,F1,G1)
06930       CALL PHDMAT(PHI2,D2,TF1,F2,G2)
06940       CALL PHDMAT(PHI3,D3,TF2,F3,G3)
06950       CALL RMMUL(TEMP1,PHI3,PHI2,3,3,3)
06960       CALL RMMUL(PHI,TEMP1,PHI1,3,3,3)
06970       CALL RMMUL(FTEMP1,PHI,X,3,3,1)
06980       CALL RMMUL(TEMP2,TEMP1,D1,3,3,2)
06990       CALL RMMUL(TEMP1,PHI3,D2,3,3,2)
07000       CALL RMADD(TEMP1,TEMP2,TEMP1,3,2)
07010       CALL RMADD(V,TEMP1,D3,3,2)
07020       CALL RMMUL(FTEMP2,V,U,3,2,1)
07030       CALL RMADD(F,FTEMP1,FTEMP2,3,1)
07040       F(2,1)=0.
07050     2 RETURN
07060       END
```

```
07070 C
07080 C
07090       SUBROUTINE GAMM1(GAM,TON,TF1,X,U,XMU)
07100       DIMENSION X(3,1),U(2,1),FBAR0(3,1),FBAR(3,1),GAM(3,1),
07110      XPHI1(3,3),D1(3,2),TEMPY1(3,1),TEMPY2(3,1),Y(3,1)
07120       COMMON/PARAM/F1(3,3),F2(3,3),F3(3,3),G1(3,2),
07130      XG2(3,2),G3(3,2)
07140       COMMON/EXTPAR/NIT,EPS,TP,ET,MODE,TF,LDUTY
07150       VARTON=TON
07160       VARTF1=TF1
07170       DELTON=XMU*TON
07180       DELTF1=XMU*TF1
07190       DELU=XMU*ABS(U(1,1))
07200       CALL FFUNC1(VARTON,VARTF1,X,U,FBAR0)
07210       U(1,1)=U(1,1)+DELU
07220       IT=0
07230       SC1=ZETA1(VARTON,X,U)
07240    67 DZETA1=(ZETA1(VARTON+DELTON,X,U)-SC1)/DELTON
07250       IT=IT+1
07260       VARTON=VARTON-SC1/DZETA1
07270       SC1=ZETA1(VARTON,X,U)
07280       IF(ABS(SC1).LT.EPS) GO TO 64
07290       IF(IT.LT.NIT) GO TO 67
07300       WRITE(6,61) IT,SC1
07310    61 FORMAT(*MAX ITERATION ON TON. IT=*,I3,* SC1=*,G12.6/)
07320       GO TO 70
07330    64 CALL PHDMAT(PHI1,D1,VARTON,F1,G1)
07340       CALL STS1(Y,PHI1,X,D1,U)
07350       IT=0
07360       IF(MODE.EQ.1) GO TO 65
07370       B1=BMAT1(VARTF1,Y,U)
07380    63 DB=(BMAT1(VARTF1+DELTF1,Y,U)-B1)/DELTF1
07390       IT=IT+1
07400       VARTF1=VARTF1-B1/DB
07410       B1=BMAT1(VARTF1,Y,U)
07420       IF(ABS(B1).LT.EPS) GO TO 65
07430       IF(IT.LT.NIT) GO TO 63
07440       WRITE(6,66)IT,B1
07450    66 FORMAT(*MAX ITERATION ON TF1. IT=*,I3,* SC1=*,G12.6/)
07460       GO TO 70
07470    65 CALL FFUNC1(VARTON,VARTF1,X,U,FBAR)
07480       DO 69 I=1,3
07490    69 GAM(I,1)=(FBAR(I,1)-FBAR0(I,1))/DELU
07500       U(1,1)=U(1,1)-DELU
07510    70 RETURN
07520       END
07530 C
07540 C
07550       SUBROUTINE SMAT1(TON,TF1,TF2,SMAT,XMU,XL0,P0)
07560       DIMENSION TEMPY1(3,1),TEMPY2(3,1),PHI1(3,3),D1(3,2),
07570      XPHI2(3,3),D2(3,2),PHI3(3,3),D3(3,2)
07580       COMMON/PARAM/ F1(3,3),F2(3,3),F3(3,3),G1(3,2),G2(3,2),
07590      XG3(3,2)
07600       COMMON/EXTPAR/NIT,EPS,TP,ET,MODE,TF,LDUTY
```

```
07610       COMMON/STATE/X(3,1),Y(3,1),Z(3,1),U(2,1)
07620       TF2=TP-TON-TF1
07630       IF(LDUTY.EQ.2)TF2=TF-TF1
07640       DELTF1=XMU*TF1
07650       CALL STATE1(TON,TF1)
07660 C     Y(3,1)=ET
07670       IT=0
07680       B1=BMAT1(TF1,Y,U)
07690       IF(ABS(B1).LT.EPS) GO TO 32
07700    31 B2=BMAT1(TF1+DELTF1,Y,U)
07710       DBMAT1=(B2-B1)/DELTF1
07720       TF1=TF1-B1/DBMAT1
07730       B1=BMAT1(TF1,Y,U)
07740       IF(ABS(B1).LT.EPS) GO TO 32
07750       IT=IT+1
07760       IF(IT.LT.NIT) GO TO 31
07770       WRITE(6,33) IT,TF1,B1
07780    33 FORMAT(*MAX. ITERATION ON IT=*,I3,*TF1=*,G15.6,* B1=*,G15.6/)
07790       TF1=SQRT(2.*XLO*TP*PO*(U(1,1)-U(2,1))/(U(1,1)*U(2,1)**2))
07800       IF(LDUTY.EQ.2)TF1=TF
07810    32 CALL PHDMAT(PHI2,D2,TF1,F2,G2)
07820       CALL STS(Z,PHI2,Y,D2,U)
07830 C     Z(2,1)=0.
07840       TF2=TP-TON-TF1
07850       IF(LDUTY.EQ.2)TF2=TF-TF1
07860       CALL PHDMAT(PHI3,D3,TF2,F3,G3)
07870       SMAT=X(3,1)-PHI3(3,1)*Z(1,1)-PHI3(3,2)*Z(2,1)-
07880      XPHI3(3,3)*Z(3,1)-D3(3,2)*U(2,1)-D3(3,1)*U(1,1)
07890       RETURN
07900       END
```

```
07910 C
07920 C
07930       FUNCTION XMAT1(TON,EI,ER)
07940       DIMENSION PHIP(3,3),PHIF(3,3),PHIN(3,3),DF(3,2),DN(3,2),
07950      X TEMP1(2,2),X(3),Y(3),TVEC1(3)
07960       COMMON/PARAM/F1(3,3),F2(3,3),F3(3,3),G1(3,2),G2(3,2),
07970      X G3(3,2)
07980       COMMON/EXTPAR/NIT,EPS,TP,ET,MODE,TF,LDUTY
07990       TF1=TP-TON
08000       IF(LDUTY.EQ.2)TF1=TF
08010       CALL PHDMAT(PHIP,DF,TP,F1,G1)
08020       CALL PHDMAT(PHIF,DF,TF1,F1,G1)
08030       CALL PHDMAT(PHIN,DN,TON,F1,G1)
08040       DO 12 I=1,2
08050       DO 11 J=1,2
08060    11 TEMP1(I,J)=-PHIP(I,J)
08070    12 TEMP1(I,I)=1.+TEMP1(I,I)
08080       DET=TEMP1(1,1)*TEMP1(2,2)-TEMP1(2,1)*TEMP1(1,2)
08090       IF(ABS(DET).LT.1.E-8) GO TO 100
08100       TVEC1(1)=PHIF(1,1)*DN(1,1)+PHIF(1,2)*DN(2,1)
08110       TVEC1(2)=PHIF(2,1)*DN(1,1)+PHIF(2,2)*DN(2,1)
08120       X(1)=EI*(TEMP1(2,2)*TVEC1(1)-TEMP1(1,2)*TVEC1(2))/DET
08130       X(2)=EI*(TEMP1(1,1)*TVEC1(2)-TEMP1(2,1)*TVEC1(1))/DET
08140       X(3)=ET-PHIN(3,1)*X(1)-PHIN(3,2)*X(2)-DN(3,1)*EI-DN(3,2)*ER
08150       Y(1)=PHIN(1,1)*X(1)+PHIN(1,2)*X(2)+DN(1,1)*EI
08160       Y(2)=PHIN(2,1)*X(1)+PHIN(2,2)*X(2)+DN(2,1)*EI
08170       Y(3)=ET
08180       SC1=PHIF(3,1)*Y(1)+PHIF(3,2)*Y(2)+ET+DF(3,2)*ER
08190       XMAT1=X(3)-SC1
08200   100 RETURN
08210       END
```

Appendix B. Boost Regulator Computer Program Description

The following tables list the boost converter PAS computer program nomenclautre, namelist parameters/variables, sample case data.

Table B1. Nomenclature

RIPX(4,1)	steady-state ripple
PSI(4,4)	Ψ
PSY(4,4)	Ψ for eigenvalue computation
GAM(4,1)	Γ
INT(8)	
XFP(4,1)	$\bar{X}_{old}$
PRAM(10)	[C1 C2 R3 R4 R5 RL N2 LO CO EI]^T
H(4)	[1 0 0 0]
R(4,5)	QRAL
ITBL(3)	QRAL
IVD(4)	QRAL
DELX(4,1)	$\Delta\underline{X}$
PHI1(4,4)	$\Phi 1$
PHI2(4,4)	$\Phi 2$
PHI3(4,4)	$\Phi 3$
D1(4,4)	D1
D2(4,4)	D2
D3(4,4)	D3
W(4,1)	

Table 82. Common Variables

COMMON/PARAM/

F1(4,4)
F2(4,4)
F3(4,4)
G1(4,4)
G2(4,4)
G3(4,4)

COMMON/EXTPAR/

NIT
EPS
NTERMS
MIT
TP
ET

COMMON/STATE/

X(4,1)
Y(4,1)
Z(4,1)
U(4,1)

Table B3. Namelist Variables

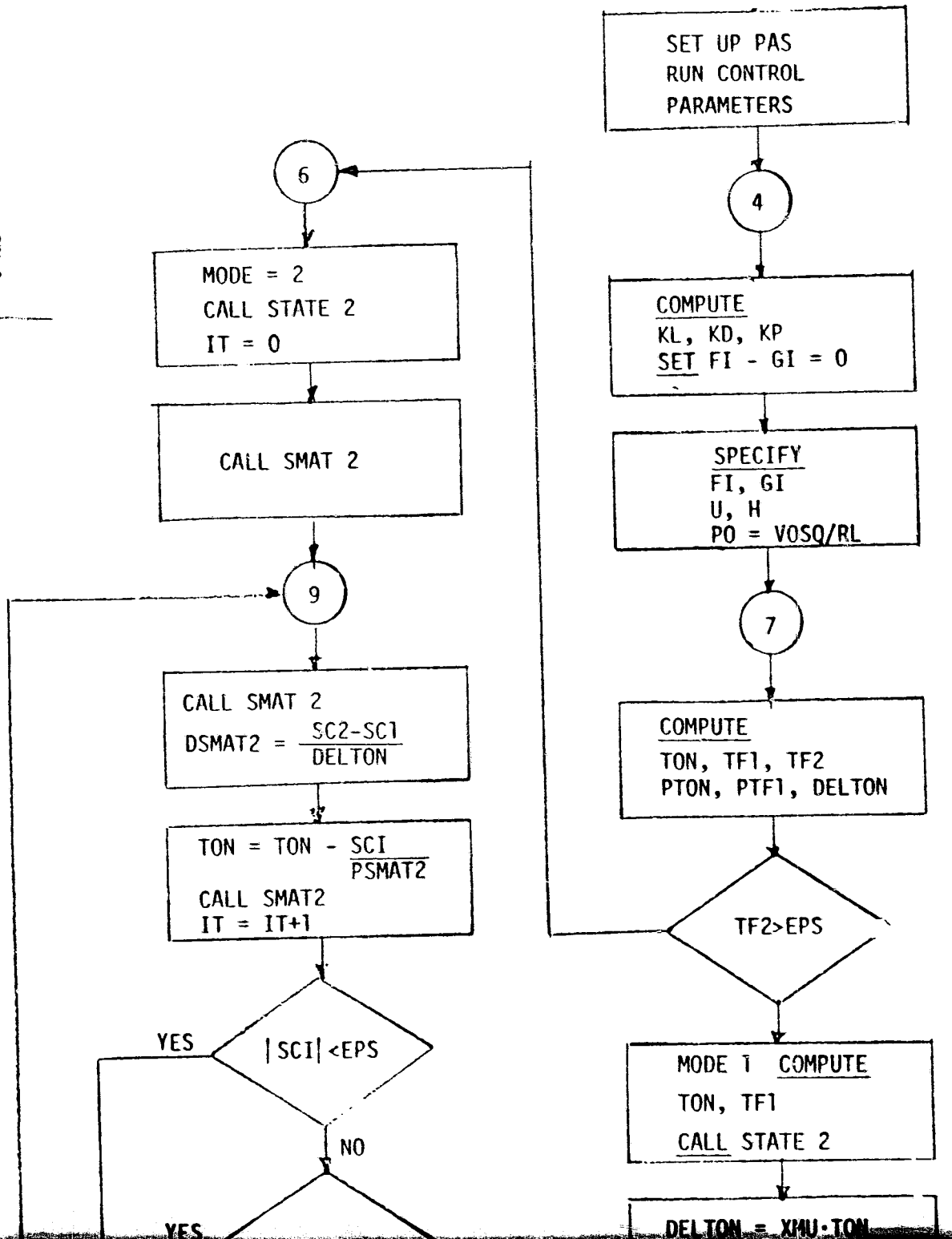
NAMELIST/PARAM/	NAMELIST/LOOP/
RO	EPS
LO	NIT
CO	XMU
RS	NTERMS
RL	MIT
R1	
R2	
R3	
R4	
R5	
C1	
C2	
N1	
N2	
EFF	
EI	
ER	
EQ	
ED	
ET	
TP	
VO	

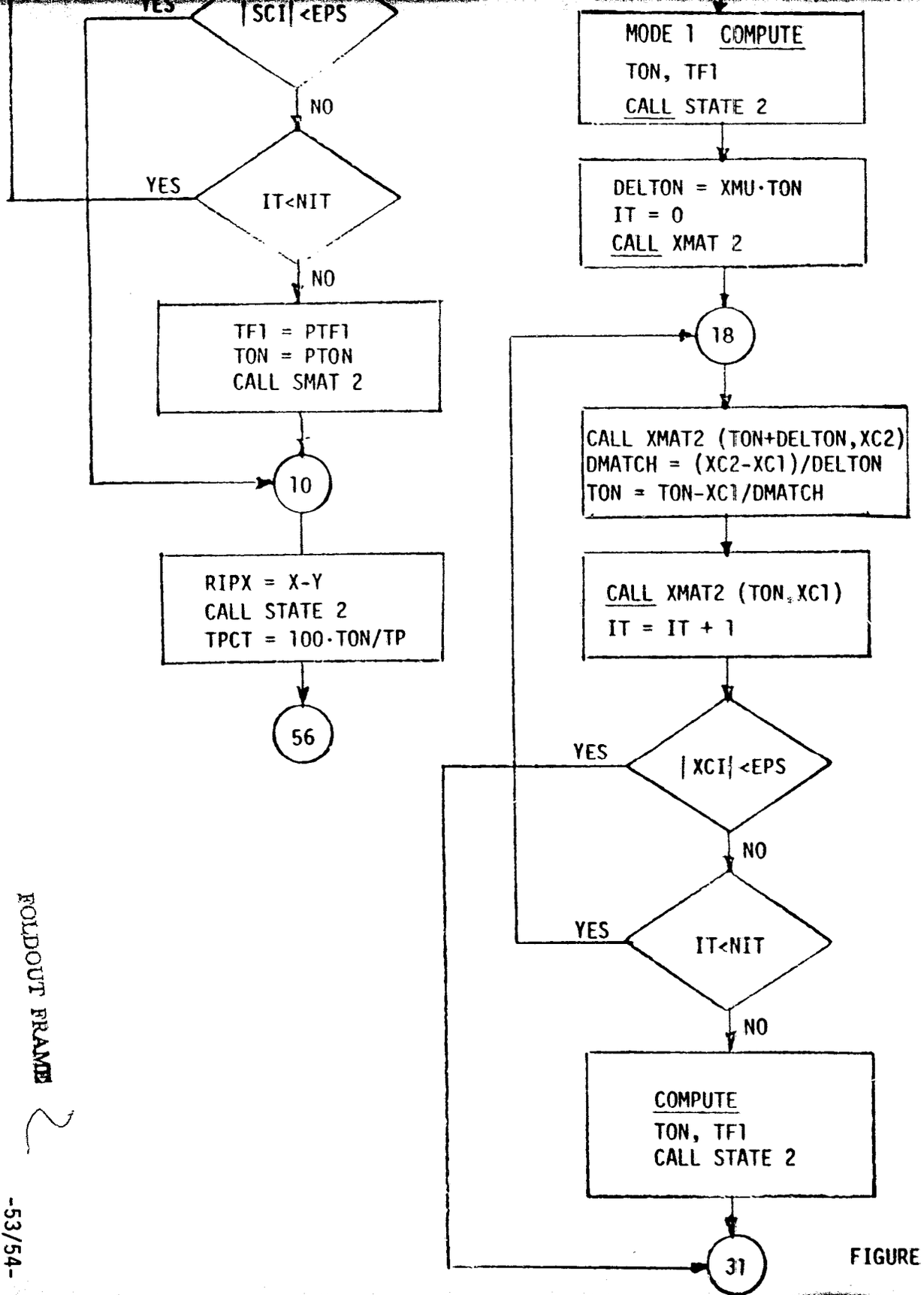
Table B4. Nominal Case Data

RO	.400	THETAO	0
LO	36.2E-6	DELTHET	5
CO	3.E-4	THETAF	180
RS	0.17		
RL	50.00	VO	37.5
R1	61.9E3	N1T	10
R2	4.9E3	N1	22
R3	40.2E3	N2	11
R4	45.1E3	EFF	0.80
R5	6.04E3		
C1	2.2E-9	NTERMS	20
C2	3.3E-9	M1T	5
		LRTL	0
EISWIT	23	NRL	2
XMU	0.01	DPRAM	0
		PRAMF	0
EI	28		
ER	2.75	LIST	1
EQ	0.2	LPEAK	0
ED	0.7	LFE	0
ET	2.5	NK	15
EPS	1.E-6	LFREQ	0
TP	33.34E-6		

The boost converter PAS computer program structure is illustrated. The flow charts for each of the computer program subroutines are presented in Figures B1-B13. Table B5 lists the boost PAS computer program.

FOLDOUT FRAME

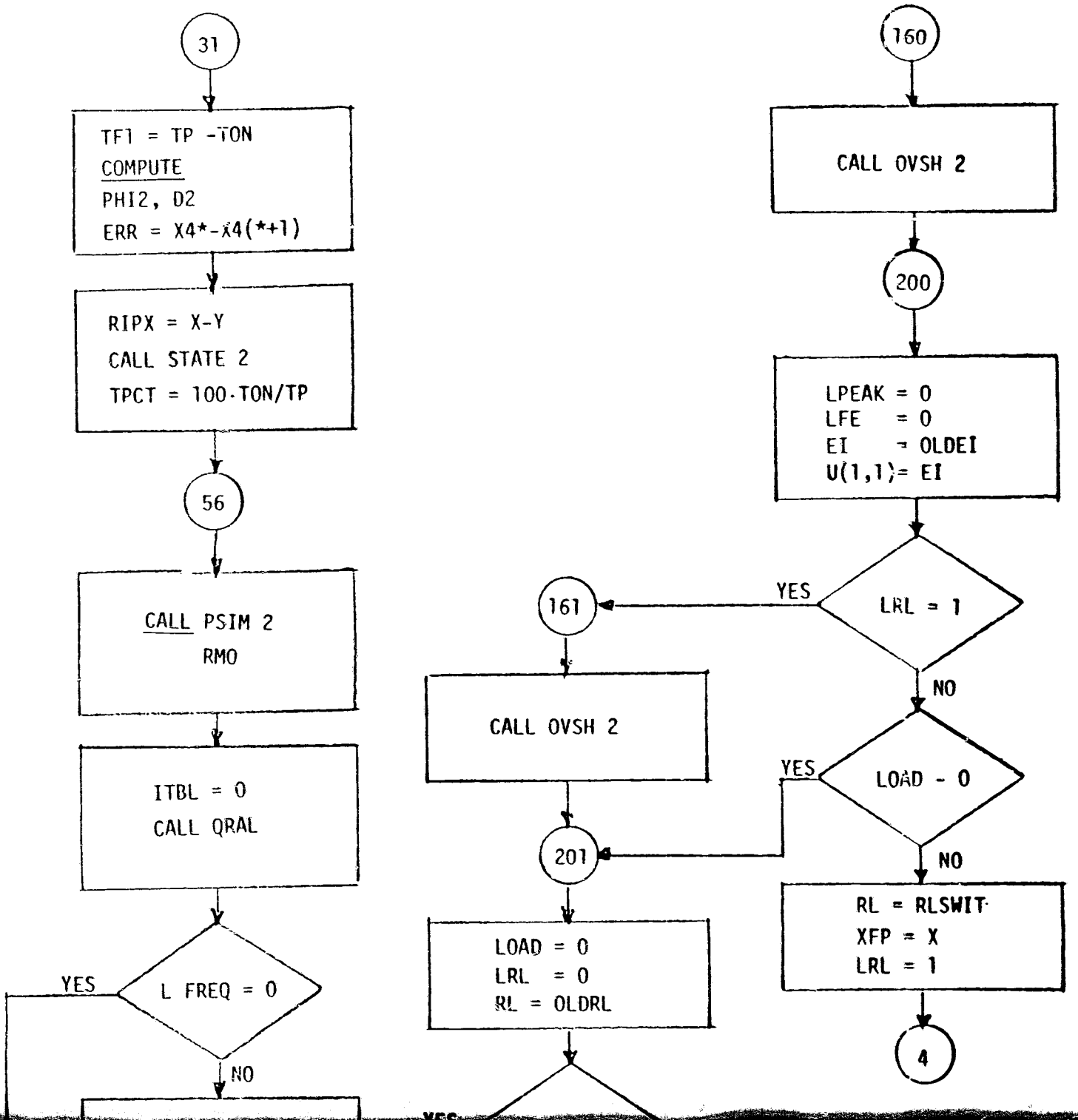




FOLDOUT FRAME 2

FIGURE B1A - PAS 2 GENERAL FLOW CHART

FOLDOUT FRAME



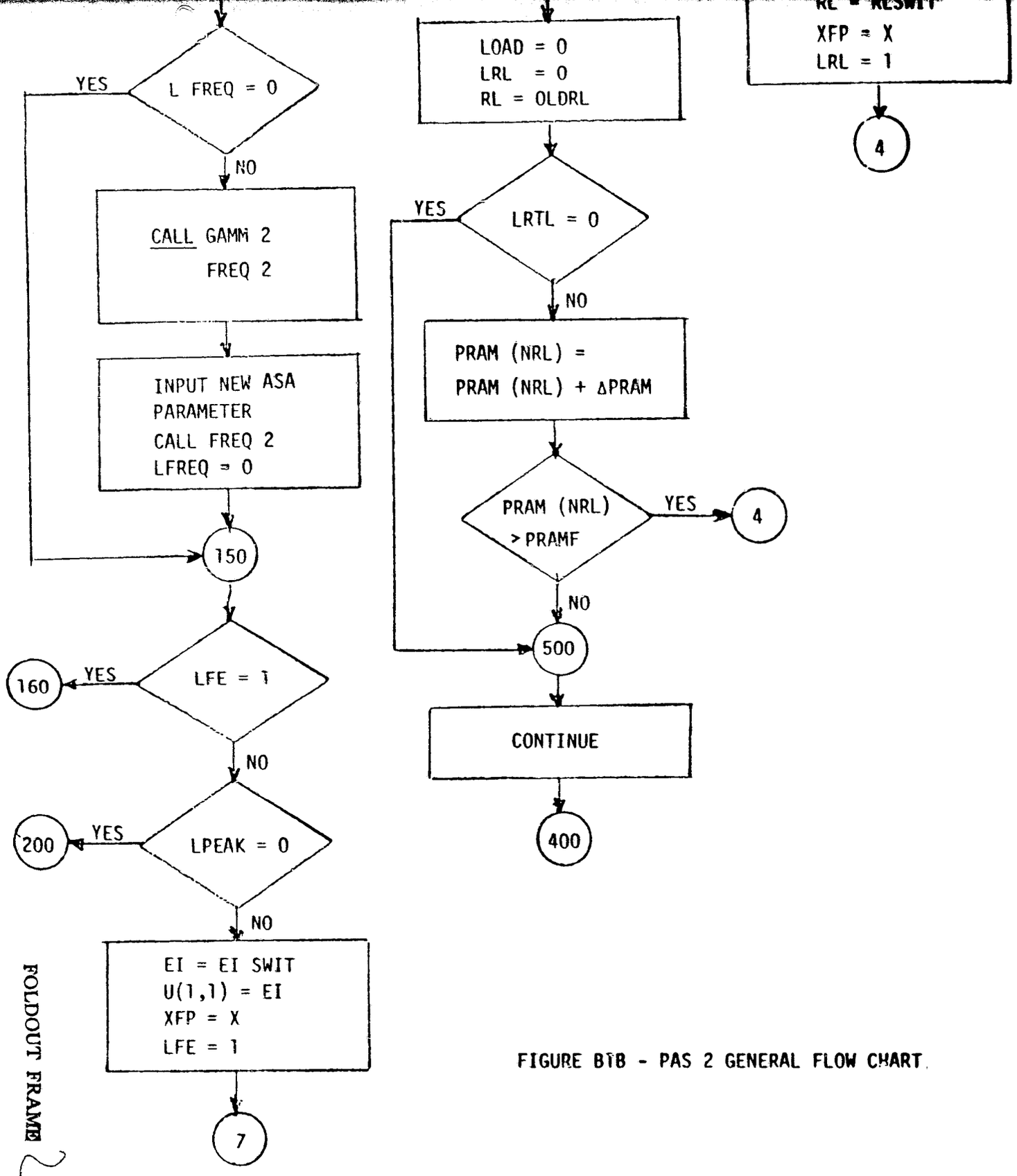


FIGURE B1B - PAS 2 GENERAL FLOW CHART

FOLDOUT FRAME 2

SUBROUTINE OVSH 2 (PSI, XFP, NK)

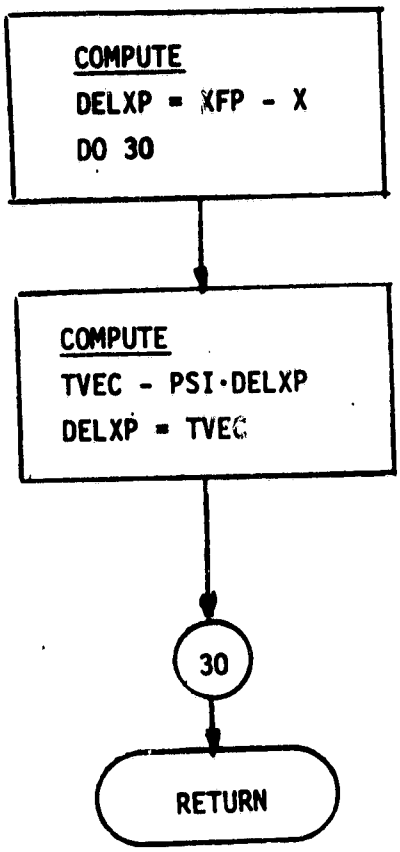


FIGURE B2 - PAS 2 SUBROUTINE OVSH 2

SUBROUTINE FREQ 2 (PSI, DVEC, H, THETAO, THETAF, DELTHET, EI, VO)

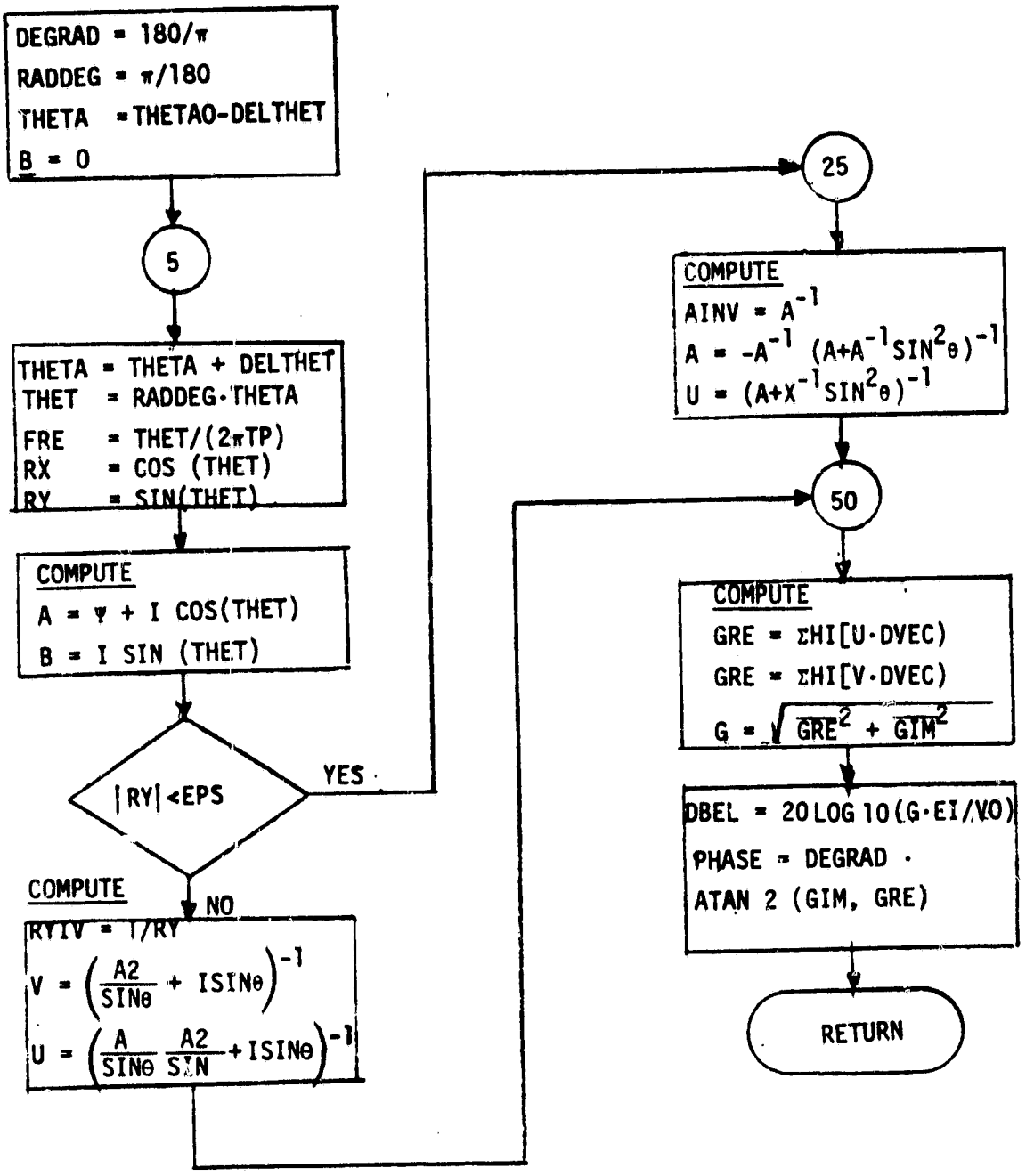
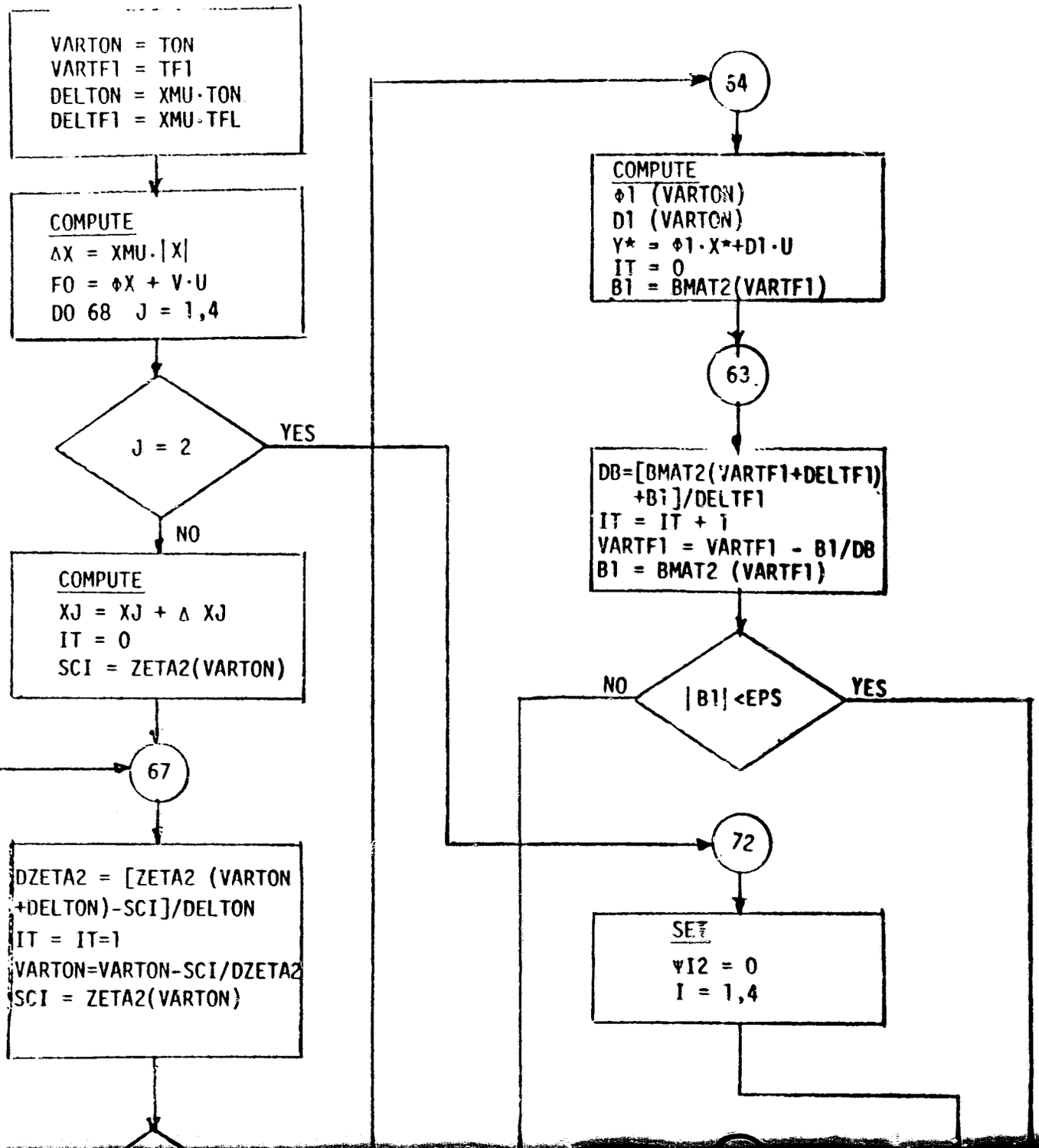


FIGURE B3 - PAS 2 SUBROUTINE

SUBROUTINE PSIM 2 (PSI, TON, TF1, X, U, XMU)

FOLDOUT FRAME



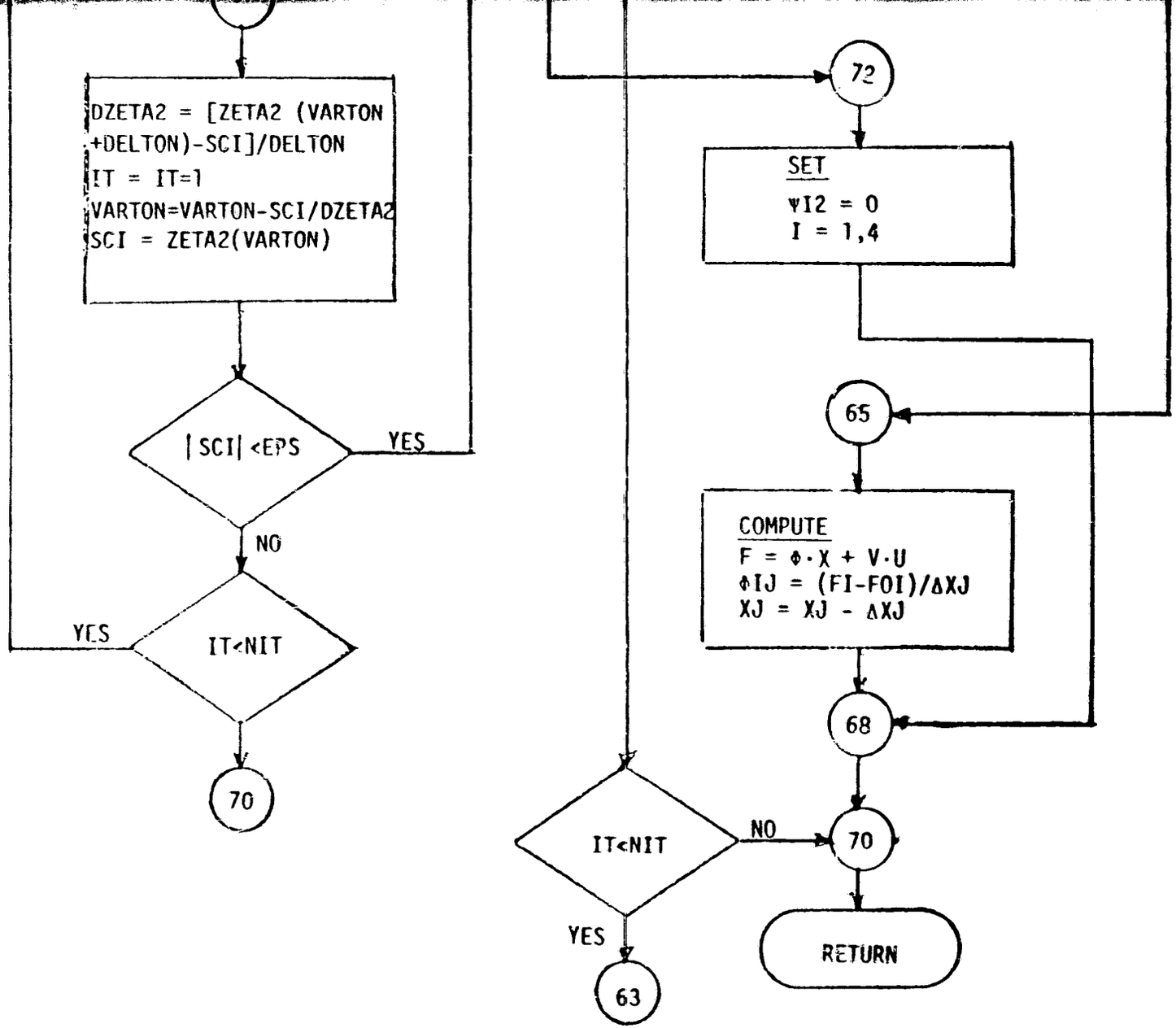


FIGURE B4 - PAS2 SUBROUTINE PSIM 2

FOLDOUT FRAME 2

SUBROUTINE DMAT 2 (T, F, G, PHI, D)

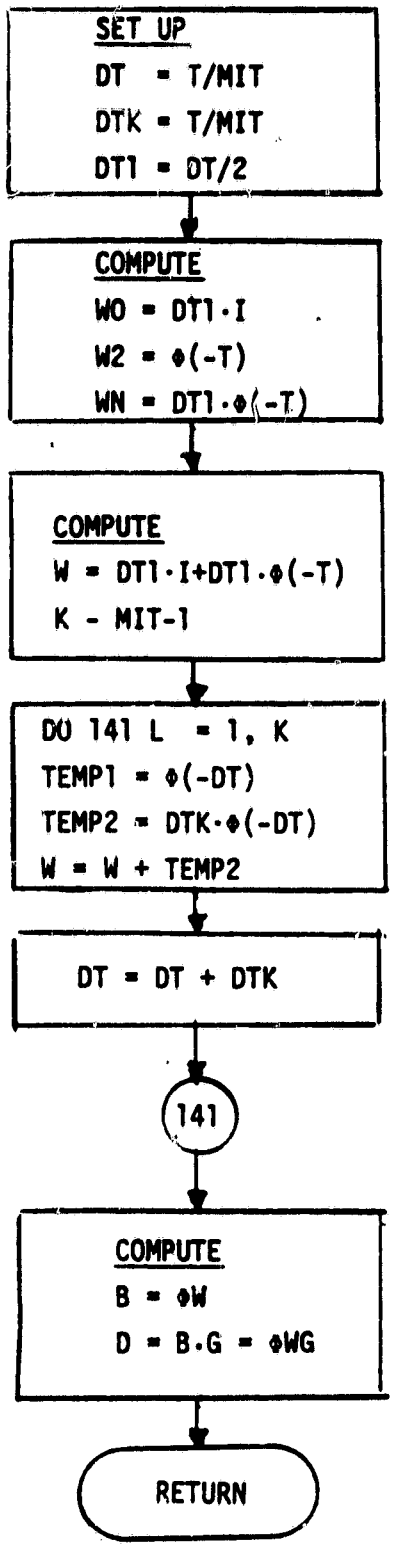


FIGURE B5 - PAS 2 SUBROUTINE DMAT 2

SUBROUTINE STATE 2 (TON:TF1)

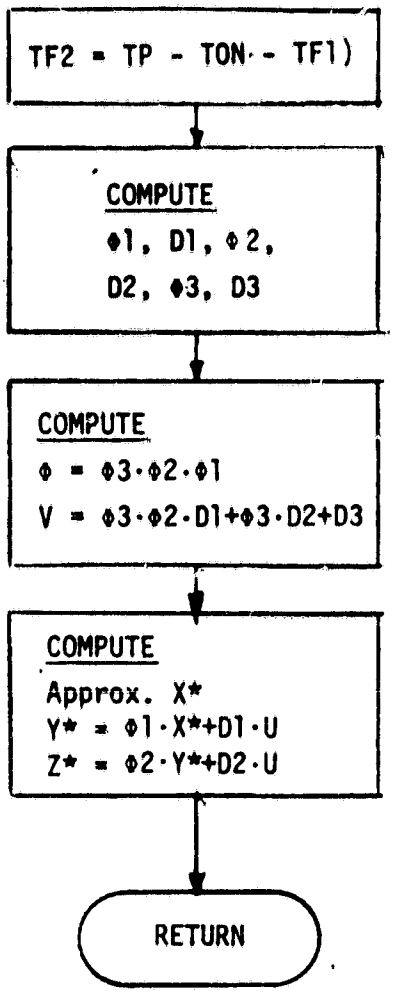


FIGURE B6 - PAS 2 SUBROUTINE STATE 2

SUBROUTINE STS 2 (W2, PHI, W1, D, U)

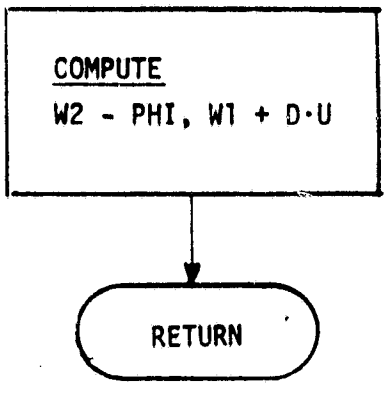


FIGURE B7 - PAS 2 SUBROUTINE STS 2

SUBROUTINE PHIM 2 (T, F, PHI)

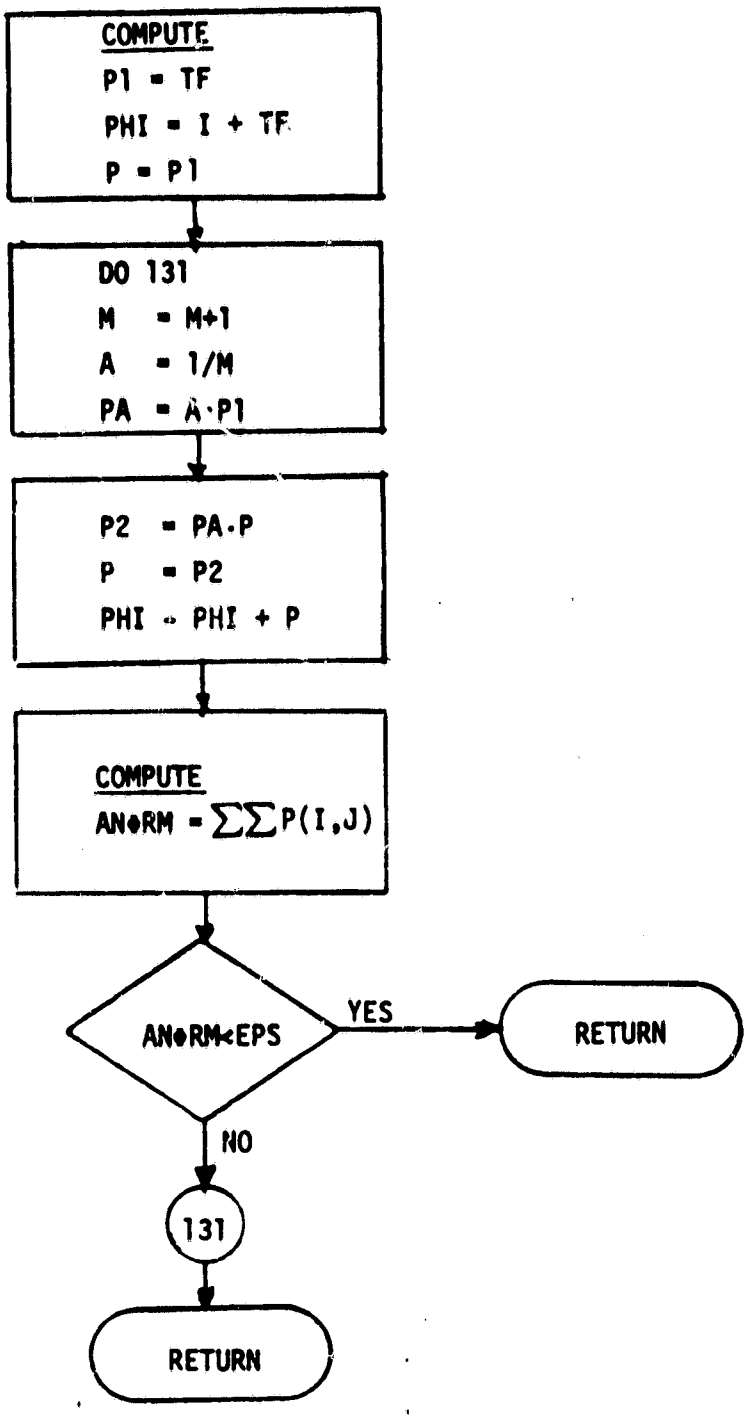


FIGURE B8 - PAS 2 SUBROUTINE PHIM 2

SUBROUTINE FFUNC 2 (TON, TF1, X, U, F)

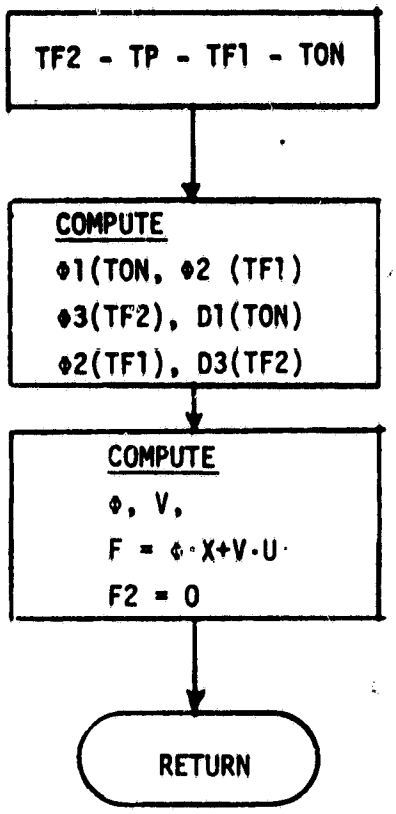


FIGURE B9 - PAS 2 SUBROUTINE FFUNC 2

SUBROUTINE GAMM 2 (TAM, TON, TF1, X, U, XMU)

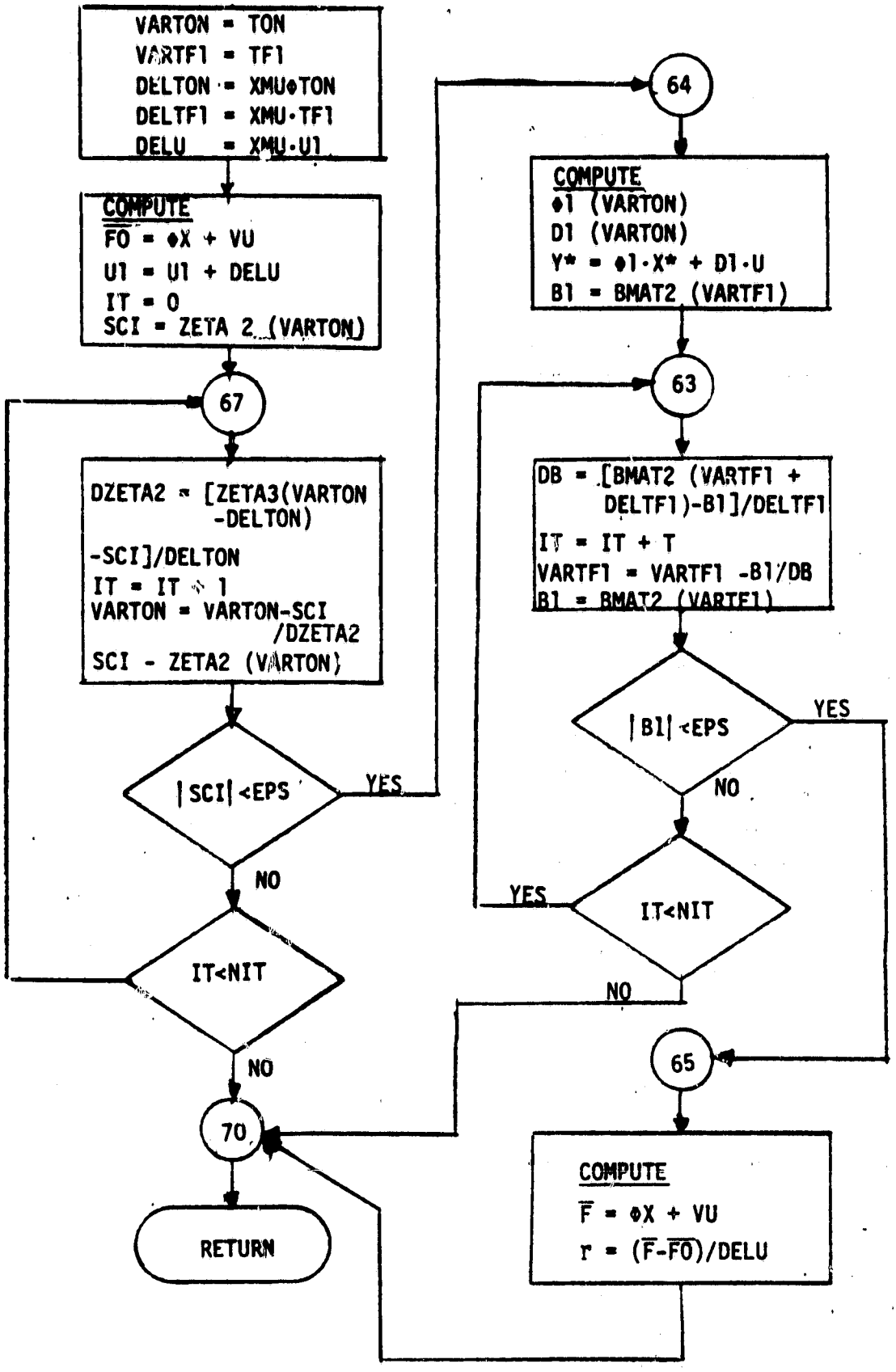


FIGURE B10 - PAS 2 SUBROUTINE GAMM 2

SUBROUTINE SMAT 2 (TON, TF1, TF2, SMAT, XMU, LO, PO, VO)

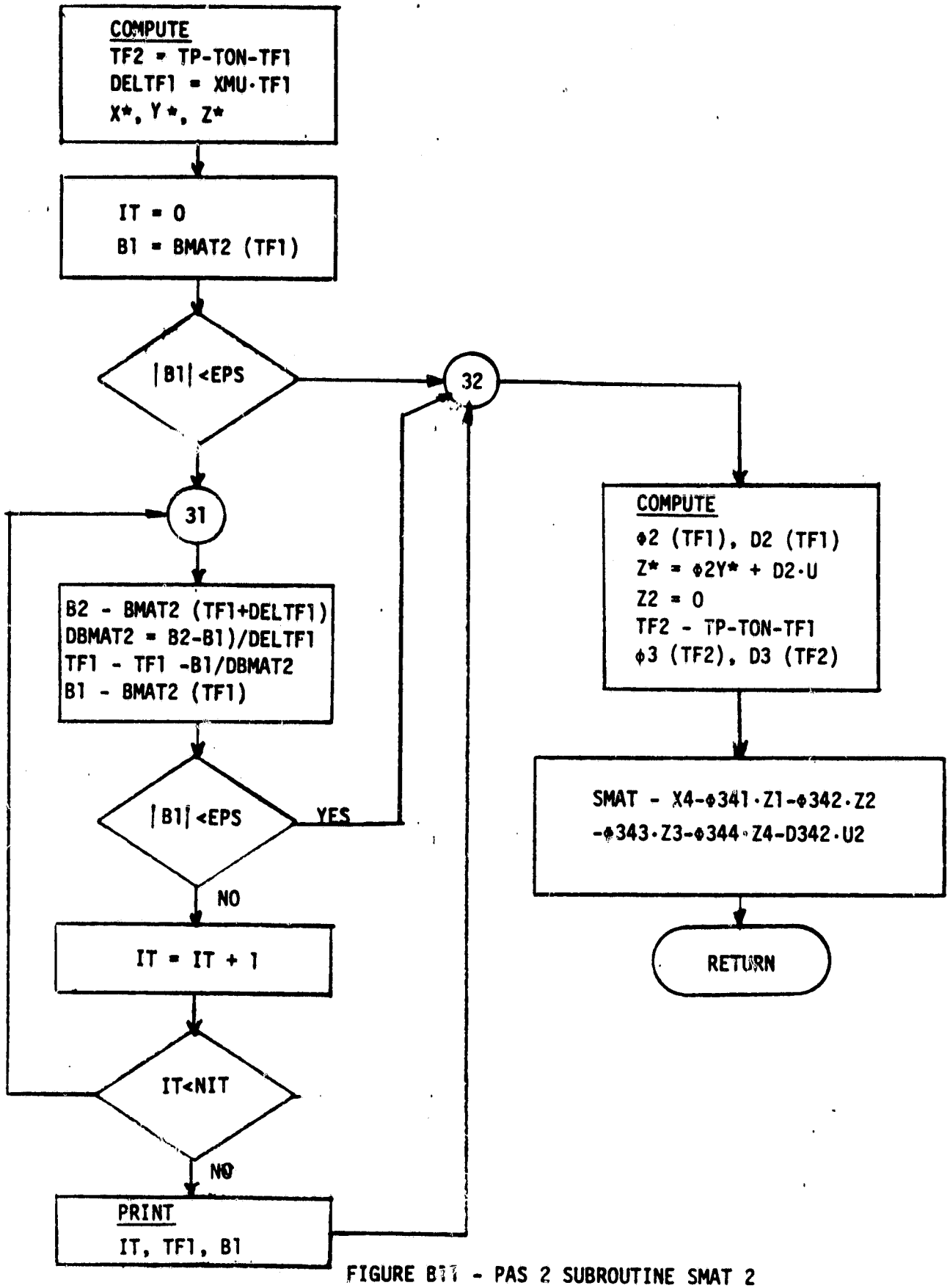


FIGURE B11 - PAS 2 SUBROUTINE SMAT 2

SUBROUTINE BMAT 2 (TF1, Y, U)

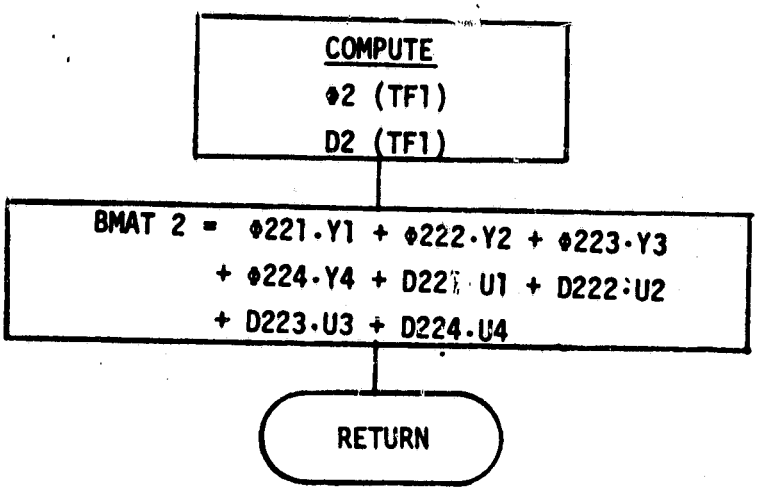


FIGURE B12 - PAS 2 SUBROUTINE BMAT 2

SUBROUTINE ZETA 2 (T, X, U)

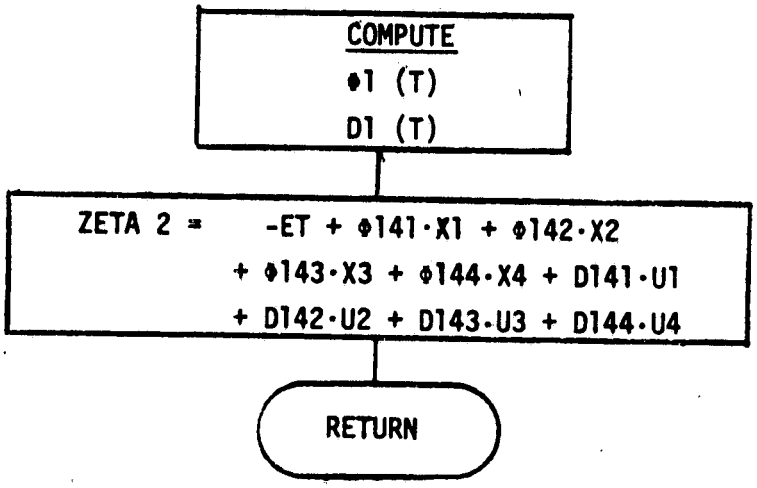


FIGURE B13 - PAS 2 SUBROUTINE ZETA 2

APPENDIX B

TABLE B5

BOOST PAS COMPUTER PROGRAM

```
00100       PROGRAM PAS2(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,
00102      XTAPE3,TAPE4,TAPE7)
00110 ****** BOOST PAS INPLEMENT TRANSIENT HISTORY ALGORITHM
00120 ****** FOR VO COMPUTATION 4.4.78
00130 ****** BOOST PAS MODIFY TF2 CUTOFF CRITERION
00140 ****** TF2.GE.TEPS=1XTP SPECIFIES MODE=2 OPERATION
00150 ****** MODIFY TRA AND LCA SEQUENCE IN ORDER TO UTILIZE
00160 ****** PROPER PSI MATRIX OPERATING POINT (EI/RL)
00170 ****** ABOVE CHANGES IMPLEMENTED 3.27.78
00180 ****** BOOST PAS FOR TP CONST (LDUTY=1) AND
00190 ****** TF CONST (LDUTY=2) 3.23.78
00200 ****** TRA DELTA VO HISTORY 3.22.78
00210 ****** LCA LOAD CHANGE ANALYSIS RL TO RLSWIT 3.22.78
      ******TAPE4 IS ASA DATA
      ******TAPE3 IS TRA DATA
00220       DIMENSION RIPX(4,1),PSI(4,4),PSY(4,4),GAM(4,1),INT(8),
00230      XXFP(4,1),PRAM(10),H(4),R(4,5),ITBL(3),IVD(4),DELX(4,1),
00240      XPHI1(4,4),PHI2(4,4),PHI3(4,4),D1(4,4),D2(4,4),D3(4,4),
00250      XW(4,1),RPRAM(10),XTR(4,1),YTR(4,1),ZTR(4,1)
00260       COMMON /PARAM/F1(4,4),F2(4,4),F3(4,4),G1(4,4),G2(4,4),
00270      XG3(4,4)
00280       COMMON/EXTPAR/NIT,EPS,NTERMS,MIT,TP,TF,LDUTY,ET,MODE,XMU
00290       COMMON /STATE/X(4,1),Y(4,1),Z(4,1),U(4,1)
00300       EQUIVALENCE (PRAM(1),C1),(PRAM(2),C2),
00310      X(PRAM(3),R3),(PRAM(4),R4),(PRAM(5),R5),
00320      X(PRAM(6),RL),(PRAM(7),N2),(PRAM(8),LO),
00330      X(PRAM(9),CO),(PRAM(10),EI)
00340       REAL LO,N1,N2,KD,KL,N,LP
00350       DATA RO,LO,CO,RS,RL/.400,36.2E-6,3.E-4,.17,50.00/
00360       DATA R1,R2,R3,R4,R5,C1,C2/61.9E3,4.9E3,40.2E3,22.55E3,
00370      X6.04E3,2.2E-9,3.3E-9/
00380       DATA EISWIT,XMU/23.,0.01/
00390       DATA EI,ER,EQ,ED,ET,EPS/30.,2.75,0.2,0.7,2.5,1.E-6/
00400       DATA TP/33.34E-6/
00410       DATA THETAO,DELTHET,THETAF/0.,.5,5./
00420       DATA VO/37.5/
00430       DATA NIT,N1,N2,EFF/50,22.,11.,.94/
00440       DATA NTERMS,MIT,LRTL,NRL,DPRAM,PRAMF/20,5,0,10,4.,40./
00450       DATA LIST,LPEAK,LFE,NK,LFREQ/1,0,0,60,0/
00460       DATA LPARAM,LCOMP,LPC,LSA,LRLPC,LCFR,LPEAK/0,0,0,0,0,0,0/
00470       DATA RPRAM/2HC1,2HC2,2HR3,2HR4,2HR5,2HRL,2HN2,2HLO,
00480      X 2HCO,2HEI/
00490       DATA RLSWIT,LRL/600.,0./
00500       DATA TF,LDUTY,TFCON/1.514365E-5,1,3.E-5/
00510       DATA OLDTIME,LTR,LRESP,TSWIT,TFINAL/0.,1,0,2.E-3,13.9E-3/
00520       DATA MTR/1/
00530       NAMELIST/PARAM/RO,LO,CO,RS,RL,R1,R2,R3,R4,R5,C1,C2,
00540      XN1,N2,EFF,EI,ER,EQ,ED,ET,TF,TF,VO
```

```
00550       NAMELIST/COMP/EPS,NIT,XMU,NTERMS,MIT
00560       NAMELIST/CONTRL/LPARAM,LCOMP,LPC,LSA,LRTL,LRLPC,LFREQ
00570      X,LCFR,LPEAK,LOAD,LDUTY,LRESP
00580       NAMELIST/RLPARAM/NRL,DPRAM,PRAMF
00590       NAMELIST/FRPARAM/THETA0,DELTHET,THETAF
00600       NAMELIST/TAPARAM/EI,EISWIT,NK
00610       NAMELIST/RPARAM/RL,RLSWIT,NK
00620       NAMELIST/TRPARAM/OLDTIME,TSWIT,TFINAL
            REWIND 3
            REWIND 4
00630       REWIND 7
00640 C     WRITE(6,99)
00650    99 FORMAT(1X,*PROGRAM FUNCTION*,13X,* CONTROL PARAMETER (1-YES 0-NO)*
00660      X /* CHANGE PARAM*,18X,*LPARAM*
00670      X /* CHANGE COMP*,19X,*LCOMP*
00680      X /* LIST PARAM, COMP*,14X,*LPC*
00690      X /* STABILITY ANALYSIS*,12X,*LSA*
00700      X /* ROOT LOCUS ANALYSIS*,11X,*LRTL*
00710      X /*       LIST PARAMETER CODE*,6X,*LRLPC*
00720      X /* AUDIO ANALYSIS*,16X,*LFREQ*
00730      X /*       CHANGE FREQ RANGE*,8X,*LCFR*
00740      X /* TRANSIENT ANALYSIS*,12X,*LPEAK*
00750      X /* TRANSIENT LOAD    *,12X,*LOAD*
00760      X /* DUTY CYCLE SCHEME *,12X,*LDUTY*
00770      X /* TRANSISTION ALGORITHM *,6X,*LRESP*/)
00780   400 CONTINUE
00790       WRITE(6,401)
00800   401 FORMAT(1X,*ENTER N TO DISCONTINUE PAS, OTHERWISE Y*)
00810       READ(5,402) X1
00820   402 FORMAT(A1)
00830       IF(X1.EQ.1HN) STOP
00840       WRITE(6,403)
00850   403 FORMAT(1X,*INPUT PAS CONTROL PARAMETERS*)
00860       READ(5,CONTRL)
00870 C     WRITE(6,CONTRL)
00880       IF(LDUTY.EQ.1) GO TO 420
00890       WRITE(6,421)
00900   421 FORMAT(/1X,*CONSTANT TF DUTY CYCLE SCHEME*/)
00910       GO TO 422
00920   420 CONTINUE
00930       WRITE(6,423)
00940   423 FORMAT(/1X,*CONSTANT TP DUTY CYCLE SCHEME*/)
00950   422 CONTINUE
00960       IF(LPARAM.EQ.0) GO TO 404
00970       READ(5,PARAM)
00980   404 IF(LCOMP.EQ.0) GO TO 405
00990       READ(5,COMP)
01000   405 IF(LPC.EQ.0) GO TO 406
01010       WRITE(6,PARAM)
01020       WRITE(6,COMP)
01030   406 IF(LSA.EQ.0) GO TO 407
```

```
01040       GO TO 5
01050   407 IF(LRTL.EQ.0) GO TO 408
01060       IF(LRLPC.EQ.0) GO TO 409
01070       WRITE(6,460)
01080   460 FORMAT(1X,*  CODE    PARAMETER*
01090      X /*   1=     C1*/*   2=     C2*/*   3=     R3*/
01100      X *   4=     R4*/*   5=     R5*/*   6=     RL*/
01110      X *   7=     N2*/*   8=     LO*/*   9=     CO*/
01120      X *  10=     EI*/)
01130   409 WRITE(6,410)
01140   410 FORMAT(1X,*INPUT ROOT LOCUS PARAMETERS*)
01150       READ(5,RLPARAM)
01160       WRITE(6,RLPARAM)
01170       GO TO 5
01180   408 IF(LFREQ.EQ.0) GO TO 411
01190       IF(LCFR.EQ.0) GO TO 5
01200       WRITE(6,413)
01210   413 FORMAT(1X,*INPUT FREQUENCY RANGE PARAMETERS*)
01220       READ(5,FRPARAM)
01230       WRITE(6,FRPARAM)
01240       GO TO 5
01250   411 IF(LPEAK.EQ.0) GO TO 415
01260       WRITE(6,414)
01270   414 FORMAT(1X,*INPUT TRANSIENT ANALYSIS PARAMETERS*)
01280       READ(5,TAPARAM)
01290       WRITE(6,TAPARAM)
01300       GO TO 5
01310   415 IF(LOAD.EQ.0) GO TO 424
01320       WRITE(6,416)
01330   416 FORMAT(1X,*INPUT LOAD CHANGE PARAMETERS*)
01340       READ(5,RPARAM)
01350       WRITE(6,RPARAM)
01360   424  CONTINUE
01370       IF(LRESP.EQ.0) GO TO 400
01380       WRITE(6,425)
01390   425  FORMAT(1X,*INPUT TRANSITION RESPONSE PARAMETERS*)
01400       READ(5,TRPARAM)
01410       WRITE(6,TRPARAM)
01420     5 CONTINUE
01430       IF(LRTL.EQ.0)GO TO 4
01440       WRITE(6,212)RPRAM(NRL),PRAM(NRL)
01450   212  FORMAT(//*ROOT LOCUS PARAMETER *,A2,* = *,G12.4)
01460     4  CONTINUE
01470       OLDEI=EI
01480       OLDRL=RL
01490       KL=RL/(RS+RL)
01500       KD=R2/(R1+R2)
01510       DO 11 I=1,4
01520       DO 11 J=1,4
01530       G1(I,J)=G2(I,J)=G3(I,J)=0.
01540    11 F1(I,J)=F2(I,J)=F3(I,J)=0.
```

```
01550        N=N2/N1
01560     19 F1(1,1)=F2(1,1)=F3(1,1)=-1./(C0*(RL+RS))
01570        F1(2,2)=-R0/L0
01580        F1(3,1)=F2(3,1)=F3(3,1)=RL/(C2*R5*(RS+RL))
01590        F1(3,3)=F2(3,3)=F3(3,3)=-1./(C2*R5)
01600        F1(4,1)=(-R2/(C1*R3*(R1+R2))-1./(C1*R5))*RL/(RS+RL)
01610        F3(4,1)=F1(4,1)
01620        F1(4,2)=N*R0/(C1*R4)
01630        F1(4,3)=F2(4,3)=F3(4,3)=1./(C1*R5)
01640        G1(2,1)=G2(2,1)=1./L0
01650        G1(2,3)=G2(2,4)=-1./L0
01660        G1(4,1)=G2(4,1)=-N/(C1*R4)
01670        G1(4,2)=G2(4,2)=G3(4,2)=1./(C1*R3)
01680        G1(4,3)=G2(4,4)=N/(C1*R4)
01690        F2(1,2)=RL/(C0*(RS+RL))
01700        F2(2,1)=-RL/(L0*(RS+RL))
01710        F2(2,2)=-R0/L0-RS*RL/(L0*(RS+RL))
01720        F2(3,2)=F2(3,1)*RS
01730        F2(4,1)=F1(4,1)+N*RL/(C1*R4*(RS+RL))
01740        F2(4,2)=F2(4,1)*RS+F1(4,2)
01750 C      WRITE(6,93)((F1(I,J),J=1,4),I=1,4)
01760 C      WRITE(6,98)((G1(I,J),J=1,4),I=1,4)
01770 C      WRITE(6,97)((F2(I,J),J=1,4),I=1,4)
01780 C      WRITE(6,96)((G2(I,J),J=1,4),I=1,4)
01790 C      WRITE(6,95)((F3(I,J),J=1,4),I=1,4)
01800 C      WRITE(6,94)((G3(I,J),J=1,4),I=1,4)
01810 C   93 FORMAT(/*F1=*,/4(4G15.4/))
01820 C   98 FORMAT(/*G1=*,/4(4G15.4/))
01830 C   97 FORMAT(/*F2=*,/4(4G15.4/))
01840 C   96 FORMAT(/*G2=*,/4(4G15.4/))
01850 C   95 FORMAT(/*F3=*,/4(4G15.4/))
01860 C   94 FORMAT(/*G3=*/4(4G15.4/))
01870        U(1,1)=EI
01880        U(2,1)=ER
01890        U(3,1)=EQ
01900        U(4,1)=ED
01910        H(2)=H(3)=H(4)=0.
01920        H(1)=KL
01930        P0=V0**2/RL
01940        LP=L0*P0
01950        QA=EFF*EI*(EI-EQ)*(V0+ED-EQ)
01960        QB=-2.*LP*(V0+ED-EI)
01970        QC=QB*TF
01980     7  CONTINUE
01990        IF(LDUTY.EQ.1) GO TO 41
02000        QD=SQRT(QB**2-4.*QA*QC)
02010        TON=(QD-QB)/(2.*QA)
02020        TF1=TON*(EI-EQ)/(V0+ED-EI)
02030        TF2=TF-TF1
02040        TP=TON+TF
02050        GO TO 42
```

```
02060   41  CONTINUE
02070       QE=-QB*TP/QA
02080       TON=SQRT(QE)
02090       TF1=TON*(EI-EQ)/(V0+ED-EI)
02100       TF2=TP-TON-TF1
02110    42 CONTINUE
02120       TEPS=.01*TP
02130       DELTON=XMU*TON
02140       IF(TF2.GE.TEPS) GO TO 6
02150       MODE=1
02160       IF(LDUTY.EQ.1) GO TO 43
02170       TON=TF*(V0+ED-EI)/(EI-EQ)
02180       TF1=TF
02190       TP=TON+TF1
02200       GO TO 44
02210    43 TON=TP*(V0+ED-EI)/(V0+ED-EQ)
02220       TF1=TP-TON
02230    44 CONTINUE
02240       OTON=TON
02250       OTF1=TF1
02260       CALL STATE2(TON,TF1)
02270 C     WRITE(6,27)TON,TF1,TF2,X,Y,Z
02280    27 FORMAT(*MODE=1*/*APPROXIMATE STEADY STATE*/*TON=*,G15.4,
02290      X *  TF1=*,G15.4,*  TF2=*,G15.4/*X=*,4G15.4/*Y=*,4G15.4/
02300      X *Z=*,4G15.4)
02310       DELTON=XMU*TON
02320       IT=0
02330       CALL XMAT2(TON,XC1)
02340   18  CALL XMAT2(TON+DELTON,XC2)
02350       DMATCH=(XC2-XC1)/DELTON
02360       TON=TON-XC1/DMATCH
02370       CALL XMAT2(TON,XC1)
02380       IT=IT+1
02390       IF(ABS(XC1).LE.EPS) GO TO 31
02400       IF(IT.LT.NIT) GO TO 18
02410       TON=OTON
02420       TF1=OTF1
02430       CALL STATE2(TON,TF1)
02440       WRITE(6,34)
02450    34 FORMAT(/*EXCEED MAX. ITERATION FOR THE EXACT STATE*/
02460      X*APPROXIMATE STATE IS CALCULATED*/)
02470    31 CONTINUE
02480       IF(LDUTY.EQ.1) GO TO 45
02490       TF1=TF
02500       TP=TON+TF1
02510       GO TO 46
02520    45 CONTINUE
02530       TF1=TP-TON
02540    46 CONTINUE
02550       CALL PHIM2(TF1,F2,PHI2)
02560       CALL DMAT2(TF1,F2,G2,PHI2,D2)
```

```
02570        ERR=X(4,1)-PHI2(4,1)*Y(1,1)-PHI2(4,2)*Y(2,1)-
02580       X PHI2(4,3)*Y(3,1)-Y(4,1)-
02590       X D2(4,1)*U(1,1)-D2(4,2)*U(2,1)-D2(4,4)*U(4,1)
02600        DO 32 I=1,4
02610    32  RIPX(I,1)=X(I,1)-Y(I,1)
02620        CALL STATE2(TON,TF1)
02630        TPCT=100.*TON/TP
02640 C      IF(LRTL.EQ.2) GO TO 36
02650        IF(MTR.EQ.2) GO TO 36
02660        WRITE(6,35) EI,RL,LDUTY,MODE,TON,TF1,TP,TPCT,X,Y,RIPX,
02670       X ERR,IT,TEPS
02680    35  FORMAT(/*EI  =*,G12.4,*  RL=*,G12.4,*  LDUTY=*,I3,
02690       X *  MODE=*,I3/*TON =*,G12.4,*  TF1=*,G12.4,*  TP=*,G12.4,
02700       X *  TPCT=*,G12.4/*X   =*,4G12.4/*Y   =*,4G12.4/
02710       X *RIPX=*,4G12.4/*ERR =*,G12.4,*  IT=*,I3,*  TEPS=*,G12.4/)
02720    36  CONTINUE
02730        GO TO 56
02740     6  CONTINUE
02750        PTON=TON
02760        PTF1=TF1
02770        MODE=2
02780        CALL STATE2(TON,TF1)
02790 C      WRITE(6,17)TON,TF1,TF2,X,Y,Z
02800    17  FORMAT(*MODE=2*/*APPROXIMATE STEADY STATE*/*TON=*,G15.4,
02810       X*  TF1=*,G15.4,*  TF2=*,G15.4/
02820       X*X=*,4G15.4/*Y=*,4G15.4/*Z=*,4G15.4/)
02830        IT=0
02840        CALL SMAT2(TON,TF1,SC1,BC1)
02850     9  CALL SMAT2(TON+DELTON,TF1,SC2,BC1)
             SERR=SC2-SC1
             ABSERR=ABS(SERR)
             IF(ABSERR.LT.1.E-6*EPS)GO TO 10
02860        DSMAT2=(SC2-SC1)/DELTON
02870        TON=TON-SC1/DSMAT2
02880        CALL SMAT2(TON,TF1,SC1,BC1)
02890        IT=IT+1
02900        IF(ABS(SC1).LE.EPS) GO TO 10
02910        IF(IT.LT.NIT) GO TO 9
02920        TF1=PTF1
02930        TON=PTON
02940        CALL SMAT2(TON,TF1,SC1,BC1)
02950    10  CONTINUE
02960        IF(LDUTY.EQ.1) GO TO 47
02970        TF2=TF-TF1
02980        TP=TON+TF
02990        GO TO 48
03000    47  CONTINUE
03010        TF2=TP-TF1-TON
03020    48  CONTINUE
03030    13  DO 12 I=1,3
03040    12  RIPX(I,1)=X(I,1)-Y(I,1)
```

```
03050        CALL STATE2(TON,TF1)
03060        TPCT=100.*TON/TP
03070 C      IF(LRTL.EQ.2) GO TO 56
03080        IF(MTR.EQ.2) GO TO 36
03090        WRITE(6,55) EI,RL,LDUTY,MODE,TON,TF1,TF2,TP,X,Y,Z,
03100       X RIPX,SC1,IT,TPCT,BC1,TEPS,SERR
03110     55 FORMAT(/*EI  =*,G12.4,*  RL=*,G12.4,*  LDUTY=*,I3,
03120       X *  MODE=*,I3,/*TON =*,G12.4,*  TF1=*,G12.4,*  TF2=*,G12.4,
03130       X *  TP=*,G12.4/*X   =*,4G12.4/*Y   =*,4G12.4/*Z   =*,
03140       X 4G12.4/*RIPX=*,4G12.4/*SC1 =*,G12.4,*  IT=*,I3,
03150       X *  TPCT=*,G12.4,*  BC1=*,G12.4/*TEPS=*,G12.4,*  SERR=*,G12.4/)
03160
03170     56 CONTINUE
03180        CALL PSIM2(PSI,TON,TF1,X,U)
03190        CALL RMCPY(PSY,PSI,4,4)
03200        ITBL(1)=4
03210        ITBL(3)=0
03220        CALL ORAL(PSY,R,4,H,V,INT,IVD,ITBL)
03230 C      IF(LRTL.NE.2) GO TO 67
03240 C      WRITE(6,68) MODE,TPCT
03250 C 67   CONTINUE
03260 C 68   FORMAT(*MODE =*,I2,* DUTY CYCLE =*,F5.2/)
03270 C      IF (LRTL.EQ.2) GO TO 72
03280        WRITE(6,70) ((PSI(I,J),J=1,4),I=1,4)
03290     70 FORMAT(*PSI=*,/4(4G15.4/))
03300        IF(MTR.EQ.2) GO TO 76
03310     72 WRITE(6,74)((R(I,J),J=1,2),I=1,4)
03320     74 FORMAT(5X,4HREAL,11X,4HIMAG,11X,4HREAL,11X,4HIMAG,
03330       X /2(4G15.4/))
03340    76  CONTINUE
03350        IF(LRESP.EQ.0) GO TO 550
03360        CALL PHIM2(TON,F1,PHI1)
03370        CALL DMAT2(TON,F1,G1,PHI1,D1)
03380        CALL PHIM2(TF1,F2,PHI2)
03390        CALL DMAT2(TF1,F2,G2,PHI2,D2)
03400        IF(MODE.EQ.1) GO TO 589
03410        CALL PHIM2(TF2,F3,PHI3)
03420        CALL DMAT2(TF2,F3,G3,PHI3,D3)
03430   589  CONTINUE
03440        TIME=OLDTIME
03450        IF(LTR.EQ.2) GO TO 501
03460        IF(MODE.EQ.1) GO TO 500
03470        CALL RMCPY(ZTR,Z,4,1)
03480        VOTR=KL*ZTR(1,1)-VO
03490 C      WRITE(6,599)TIME,ZTR,VOTR
03500        WRITE(7,599)TIME,ZTR,VOTR
03510   599  FORMAT(6G12.4)
03520        GO TO 501
03530   500  CONTINUE
03540        CALL RMCPY(YTR,Y,4,1)
03550        VOTR=KL*YTR(1,1)+RS*KL*YTR(2,1)-VO
```

```
C     WRITE(6,599)TIME,YTR,VOTR
      WRITE(7,599)TIME,YTR,VOTR
501   CONTINUE
      TIME=TIME+TON
      IF(MODE.EQ.1) GO TO 502
      CALL STS2(XTR,PHI3,ZTR,D3,U)
      GO TO 503
502   CONTINUE
      CALL STS2(XTR,PHI2,YTR,D2,U)
503   CONTINUE
      VOTR=KL*XTR(1,1)-VO
C     WRITE(6,599)TIME,XTR,VOTR
      WRITE(7,599)TIME,XTR,VOTR
      TIME=TIME+TF1
      CALL STS2(YTR,PHI1,XTR,D1,U)
      VOTR=KL*YTR(1,1)+RS*KL*YTR(2,1)-VO
C     WRITE(6,599)TIME,YTR,VOTR
      WRITE(7,599)TIME,YTR,VOTR
      IF(MODE.EQ.1) GO TO 504
      TIME=TIME+TF2
      CALL STS2(ZTR,PHI2,YTR,D2,U)
      VOTR=KL*ZTR(1,1)-VO
C     WRITE(6,599)TIME,ZTR,VOTR
      WRITE(7,599)TIME,ZTR,VOTR
504   CONTINUE
      IF(TIME.GT.TFINAL) GO TO 505
      IF(TIME.LT.TSWIT) GO TO 501
      IF(LTR.EQ.2) GO TO 501
      IF(MODE.EQ.2) GO TO 506
      CALL RMOVE(ZTR,YTR,4,1)
506   CONTINUE
      U(1,1)=EISWIT
      EI=EISWIT
      OLDTIME=TIME
      LTR=2
      GO TO 7
505   CONTINUE
      OLDTIME=0.
      LTR=1
      MTR=2
      LRESP=0
      GO TO 7
550   CONTINUE
      MTR=1
      IF(LFREQ.EQ.0) GO TO 150
      CALL GAMM2(GAM,TON,TF1,X,U)
      WRITE(6,75)  (GAM(I,1),I=1,4)
   75 FORMAT(/*GAM=*/4(G15.4/))
   69 CALL FREQ2(PSI,GAM,H,THETA0,THETAF,DELTHET,EI,VO)
      READ(5,FRPARAM)
      IF(THETAF.EQ.0.)GO TO 149
```

```
04070       WRITE(6,FRPARAM)
04080       GO TO 69
04090   149 CONTINUE
04100       LFREQ=0
04110   150 CONTINUE
04120       IF(LFE.EQ.1) GO TO 160
04130       IF(LPEAK.EQ.0) GO TO 200
04140       DO 152 I=1,4
04150   152 XFP(I,1)=X(I,1)
04160       EI=EISWIT
04170       U(1,1)=EI
04180       LFE=1
04190       WRITE(6,189)
04200   189 FORMAT(1X,*SET UP EI STEP INPUT*)
04210       GO TO 7
04220   160 CALL OVSH2(PSI,XFP,NK)
04230   200 CONTINUE
04240       LPEAK=LFE=0
04250       EI=OLDEI
04260       U(1,1)=EI
04270       IF(LRL.EQ.1) GO TO 161
04280       IF(LOAD.EQ.0) GO TO 201
04290       RL=RLSWIT
04300       LRL=1
04310       DO 153 I=1,4
04320   153 XFP(I,1)=X(I,1)
04330       WRITE(6,188)
04340   188 FORMAT(1X,*SET UP RL LOAD CHANGE*)
04350       GO TO 4
04360   161 CALL OVSH2(PSI,XFP,NK)
04370   201 CONTINUE
04380       LOAD=LRL=0
04390       RL=OLDRL
04400       IF(LRTL.EQ.0) GO TO 300
04410       PRAM(NRL)=PRAM(NRL)+DPRAM
04420       IF(PRAM(NRL).GT.PRAMF) GO TO 300
04430       LRTL=2
04440       GO TO 5
04450   300 CONTINUE
04460       GO TO 400
04470       END
04480 ************************************************************
04490       SUBROUTINE OVSH2(PSI,XFP,NK)
04500       DIMENSION PSI(4,4),XFP(4,1),DELXP(4,1),TVEC(4,1)
04510       COMMON/STATE/X(4,1),Y(4,1),Z(4,1),U(4,1)
04520       COMMON/EXTPAR/NIT,EPS,NTERMS,MIT,TP,TF,LDUTY,EI,MODE,XMU
04530       DATA RS,C0/.013,6.E-4/
04540       G=RS*C0/TP
04550       DO 10 I=1,4
04560    10 DELXP(I,1)=XFP(I,1)-X(I,1)
04570       DELV0=DELXP(1,1)+G*DELXP(1,1)
```

```
04580       DO 30 NN=1,NK
04590       WRITE(6,12) NN,DELXP,DELV0
04600       WRITE(3,12) NN,DELXP,DELV0
04610    12 FORMAT(I3,5G12.3)
04620       CALL RMMUL(TVEC,PSI,DELXP,4,4,1)
04630       DELV0=TVEC(1,1)+G*(TVEC(1,1)-DELXP(1,1))
04640       DO 14 I=1,4
04650    14 DELXP(I,1)=TVEC(I,1)
04660    30 CONTINUE
04670       RETURN
04680       END
04690 ************************************************************
04700       FUNCTION ZETA2(T,X,U)
04710       DIMENSION PHI1(4,4),D1(4,4),X(4,1),U(4,1)
04720       COMMON/PARAM/F1(4,4),F2(4,4),F3(4,4),G1(4,4),
04730      XG2(4,4),G3(4,4)
04740       COMMON/EXTPAR/NIT,EPS,NTERMS,MIT,TP,TF,LDUTY,ET,MODE,XMU
04750       CALL PHIM2(T,F1,PHI1)
04760       CALL DMAT2(T,F1,G1,PHI1,D1)
04770       ZETA2=-ET+PHI1(4,1)*X(1,1)+PHI1(4,2)*X(2,1)+
04780      XPHI1(4,3)*X(3,1)+X(4,1)+D1(4,1)*U(1,1)+
04790      XD1(4,2)*U(2,1)+D1(4,3)*U(3,1)
04800       RETURN
04810       END
04820 ************************************************************
04830       SUBROUTINE FREQ2(PSI,DVEC,H,THETA0,THETAF,DELTHET,EI,V0)
04840       DIMENSION PSI(4,4),DVEC(4,1),H(4)
04850       DIMENSION A(4,4),AINV(4,4),B(4,4),U(4,4),V(4,4),TEMP1(4,4)
04860       DIMENSION TEMP2(4,4),TVEC1(4,1),TVEC2(4,1)
04870       COMMON/EXTPAR/NIT,EPS,NTERMS,MIT,TP,TF,LDUTY,ET,MODE,XMU
04880       DEGRAD=180./3.141592
04890       RADDEG=1./DEGRAD
04900       THETA=THETA0-DELTHET
04910       DO 2 I=1,4
04920       DO 2 J=1,4
04930    2  B(I,J)=0.
04940 C     WRITE(4,59)EI,MODE
04950    59 FORMAT(*AUDIOSUSCEPTIBILITY*,* EI=*,F6.2,* MODE=*,I2)
04960       WRITE(6,1) EI
04970    1  FORMAT(//*EI=*,E12.6/,
04980      X * THETA   FREQ (HZ)   DBEL*,5X,*G*,11X,*REG*,10X,*IMG*,
04990      X 6X,*PHASE*)
05000    5  CONTINUE
05010       THETA=THETA+DELTHET
05020       THET=RADDEG*THETA
05030       FRE=THET/(6.2831853*TP)
05040       RX=COS(THET)
05050       RY=SIN(THET)
05060       DO 12 I=1,4
05070       DO 10 J=1,4
05080    10 A(I,J)=-PSI(I,J)
```

```
05090         A(I,I)=A(I,I)+RX
05100    12   B(I,I)=RY
05110         IF(ABS(RY).LT.1.E-10) GO TO 25
05120         RYIV=1./RY
05130         CALL RMMUL(TEMP1,A,A,4,4,4)
05140         CALL RMSCLR(TEMP2,RYIV,TEMP1,4,4)
05150         DO 14 I=1,4
05160    14   TEMP2(I,I)=TEMP2(I,I)+RY
05170         CALL RMINV(V,TEMP2,4)
05180         CALL RMMUL(U,A,V,4,4,4)
05190         DO 16 I=1,4
05200         DO 16 J=1,4
05210         U(I,J)=RYIV*U(I,J)
05220    16   V(I,J)=-V(I,J)
05230         GO TO 50
05240    25   CALL RMINV(AINV,A,4)
05250         DO 30 I=1,4
05260         DO 30 J=1,4
05270    30   TEMP1(I,J)=AINV(I,J)*RY**2
05280         CALL RMADD(TEMP2,A,TEMP1,4,4)
05290         CALL RMINV(U,TEMP2,4)
05300         CALL RMMUL(V,AINV,U,4,4,4)
05310         DO 34 I=1,4
05320         DO 34 J=1,4
05330    34   V(I,J)=-RY*V(I,J)
05340    50   CONTINUE
05350         CALL RMMUL(TVEC1,U,DVEC,4,4,1)
05360         CALL RMMUL(TVEC2,V,DVEC,4,4,1)
05370         GRE=GIM=0.
05380         DO 55 I=1,4
05390         GRE=GRE+H(I)*TVEC1(I,1)
05400    55   GIM=GIM+H(I)*TVEC2(I,1)
05410         G=SQRT(GRE**2+GIM**2)
05420         DBEL=20.*ALOG10(G*EI/V0)
05430         PHASE=DEGRAD*ATAN2(GIM,GRE)
05440         WRITE(6,60) THETA,FRE,DBEL,G,GRE,GIM,PHASE
05450         WRITE(4,58)FRE,DBEL,PHASE
05460     58 FORMAT(G15.4,2F12.4)
05470    60   FORMAT(F6.2,E12.4,F9.2,3E12.4,F9.2)
05480         IF(THETA.LT.THETAF-0.5*DELTHET) GO TO 5
05490   100   CONTINUE
05500         RETURN
05510         END
05520 ************************************************************
05530         SUBROUTINE PHIM2(T,F,PHI)
05540         DIMENSION P1(4,4),F(4,4),PHI(4,4),PA(4,4),P2(4,4),F(4,4)
05550         COMMON/EXTPAR/NIT,EPS,NTERMS,MIT,TP,TF,LDUTY,ET,MODE,XMU
05560         CALL RMSCLR(P1,T,F,4,4)
05570         CALL RMCPY(PHI,P1,4,4)
05580         DO 160 I=1,4
05590    160 PHI(I,I)=P1(I,I)+1.
```

```
05600       CALL RMCPY(P,P1,4,4)
05610       DO 131 M=1,NTERMS
05620       M=M+1
05630       A=1./M
05640       CALL RMSCLR(PA,A,P1,4,4)
05650       CALL RMMUL(P2,PA,P,4,4,4)
05660       CALL RMCPY(P,P2,4,4)
05670       CALL RMADD(PHI,PHI,P,4,4)
05680       DO 132 I=1,4
05690       DO 132 J=1,4
05700   132 ANORM=ANORM+ABS(P(I,J))
05710       IF (ANORM.LT.EPS) GO TO 133
05720   131 CONTINUE
05730   133 RETURN
05740       END
05750 ************************************************************
05760       SUBROUTINE DMAT2(T,F,G,PHI,D)
05770       DIMENSION TEMP1(4,4),TEMP2(4,4),W0(4,4),W2(4,4),WN(4,4),
05780      XW(4,4),F(4,4),G(4,4),B(4,4),PHI(4,4),D(4,4)
05790       COMMON/EXTPAR/NIT,EPS,NTERMS,MIT,TP,TF,LDUTY,ET,MODE,XMU
05800       DT=T/MIT
05810       DTK=T/MIT
05820       DT1=DT/2
05830       DO 136 I=1,4
05840       DO 136 J=1,4
05850   136 W0(I,J)=0.
05860       DO 137 I=1,4
05870   137 W0(I,I)=W0(I,I)+DT1
05880       CALL PHIM2(-T,F,W2)
05890       CALL RMSCLR(WN,DT1,W2,4,4)
05900       CALL RMADD(W,W0,WN,4,4)
05910       K=MIT-1
05920       DO 141 L=1,K
05930       CALL PHIM2(-DT,F,TEMP1)
05940       CALL RMSCLR(TEMP2,DTK,TEMP1,4,4)
05950       CALL RMADD(W,W,TEMP2,4,4)
05960       DT=DT+DTK
05970   141 CONTINUE
05980       CALL RMMUL(B,PHI,W,4,4,4)
05990       CALL RMMUL(D,B,G,4,4,4)
06000       RETURN
06010       END
06020 ************************************************************
06030       SUBROUTINE STATE2(TON,TF1)
06040       DIMENSION PHI1(4,4),PHI2(4,4),PHI3(4,4),D1(4,4),D2(4,4),
06050      XD3(4,4),TEMP1(4,4),TEMP2(4,4),PHI(4,4),V(4,4),VU(4,1)
06060       DIMENSION W(4,1),TV1(4,1),TV2(4,1)
06070       COMMON/PARAM/ F1(4,4),F2(4,4),F3(4,4),G1(4,4),G2(4,4),
06080      XG3(4,4)
06090       COMMON/EXTPAR/NIT,EPS,NTERMS,MIT,TP,TF,LDUTY,ET,MODE,XMU
06100       COMMON/STATE/X(4,1),Y(4,1),Z(4,1),U(4,1)
```

```
06110        CALL PHIM2(TON,F1,PHI1)
06120        CALL DMAT2(TON,F1,G1,PHI1,D1)
06130        CALL PHIM2(TF1,F2,PHI2)
06140        CALL DMAT2(TF1,F2,G2,PHI2,D2)
06150        CALL RMMUL(PHI,PHI2,PHI1,4,4,4)
06160        CALL RMMUL(TEMP1,PHI2,D1,4,4,4)
06170        CALL RMADD(V,TEMP1,D2,4,4)
06180        IF(MODE.EQ.1) GO TO 15
06190        IF(LDUTY.EQ.1) GO TO 31
06200        TF2=TF-TF1
06210        GO TO 32
06220     31 CONTINUE
06230        TF2=TP-TON-TF1
06240     32 CONTINUE
06250        CALL PHIM2(TF2,F3,PHI3)
06260        CALL DMAT2(TF2,F3,G3,PHI3,D3)
06270        CALL RMMUL(TEMP2,PHI3,PHI,4,4,4)
06280        CALL RMCPY(PHI,TEMP2,4,4)
06290        CALL RMMUL(TEMP1,PHI3,V,4,4,4)
06300        CALL RMADD(V,TEMP1,D3,4,4)
06310     15 CONTINUE
06320        CALL RMMUL(VU,V,U,4,4,1)
06330        DEN = (1.-PHI(1,1))*(1.-PHI(2,2)) - PHI(1,2)*PHI(2,1)
06340        DET = DEN * (1. - PHI(3,3))
06350        X(1,1)=((1.-PHI(2,2))*VU(1,1) + PHI(1,2)*VU(2,1))/DEN
06360        X(3,1)=((PHI(2,1)*PHI(3,2)+PHI(3,1)*(1.-PHI(2,2)))*VU(1,1)
06370       X +(PHI(3,2)*(1.-PHI(1,1))+PHI(1,2)*PHI(3,1))*VU(2,1))/DET
06380       X +VU(3,1)/(1.-PHI(3,3))
06390        IF(MODE.EQ.2) GO TO 25
06400        X(2,1)=(PHI(2,1)*VU(1,1)+(1.-PHI(1,1))*VU(2,1))/DEN
06410        GO TO 24
06420     25 X(2,1)=0.0
06430     24 CONTINUE
06440        X(4,1)=ET-PHI1(4,1)*X(1,1)-PHI1(4,2)*X(2,1)-
06450       ZPHI1(4,3)*X(3,1)-D1(4,1)*U(1,1)-D1(4,3)*U(3,1)-
06460       ZD1(4,2)*U(2,1)
06470        CALL STS2(W,PHI,X,V,U)
06480        CALL STS2(Y,PHI1,X,D1,U)
06490        IF(MODE.EQ.1) GO TO 23
06500        CALL STS2(Z,PHI2,Y,D2,U)
06510        Z(2,1)=0.
06520     23 CONTINUE
06530 C      WRITE(6,99)((PHI1(I,J),J=1,4),I=1,4)
06540 C      WRITE(6,98)((D1(I,J),J=1,4),I=1,4)
06550 C      WRITE(6,97)((PHI2(I,J),J=1,4),I=1,4)
06560 C      WRITE(6,96)((D2(I,J),J=1,4),I=1,4)
06570        IF(MODE.EQ.1)GO TO 35
06580 C      WRITE(6,89)((PHI3(I,J),J=1,4),I=1,4)
06590 C      WRITE(6,88)((D3(I,J),J=1,4),I=1,4)
06600     35 CONTINUE
06610 C      WRITE(6,95)((PHI(I,J),J=1,4),I=1,4)
```

```
06620 C      WRITE(6,94)((V(I,J),J=1,4),I=1,4)
06630 C 99   FORMAT(/*PHI1=*,/4(4G15.4/)/)
06640 C 98   FORMAT(/*D1=*,/4(4G15.4/)/)
06650 C 97   FORMAT(/*PHI2=*,/4(4G15.4/)/)
06660 C 96   FORMAT(/*D2=*,/4(4G15.4/)/)
06670 C 89   FORMAT(/*PHI3=*,/4(4G15.4/)/)
06680 C 88   FORMAT(/*D3=*,/4(4G15.4/)/)
06690 C 95   FORMAT(*PHI=*,/4(4G15.4/))
06700 C 94   FORMAT(*V=*,/4(4G15.4/))
      C      WRITE(6,79)X,W,TON,TF1,TF2
        79   FORMAT(*X =*,4G15.4/*W =*,4G15.4/
             Z*TON =*,G15.4,*  TF1 =*,G15.4,*  TF2 =*,
             ZG15.4)
06710         RETURN
06720         END
06730 ********************************************************
06740         SUBROUTINE STS2(W2,PHI,W1,D,U)
06750         DIMENSION PHI(4,4),W1(4,1),W2(4,1),D(4,4),U(4,1),
06760        XTEMPY1(4,1),TEMPY2(4,1)
06770         CALL RMMUL(TEMPY1,PHI,W1,4,4,1)
06780         CALL RMMUL(TEMPY2,D,U,4,4,1)
06790         CALL RMADD(W2,TEMPY1,TEMPY2,4,1)
06800         RETURN
06810         END
06820 ********************************************************
06830         SUBROUTINE PSIM2(PSI,TON,TF1,X,U)
06840         DIMENSION X(4,1),FBAR0(4,1),FBAR(4,1),PSI(4,4),
06850        XU(4,1),Y(4,1),PHI1(4,4),D1(4,4),
06860        XTEMPY1(4,1),TEMPY2(4,1),DELTX(4,1)
06870         COMMON/PARAM/F1(4,4),F2(4,4),F3(4,4),G1(4,4),
06880        XG2(4,4),G3(4,4)
06890         COMMON/EXTPAR/NIT,EPS,NTERMS,MIT,TP,TF,LDUTY,ET,MODE,XMU
06900         VTON=TON
06910         VTF1=TF1
06920         DELTON=XMU*TON
06930         DELTF1=XMU*TF1
06940         DO 71 I=1,4
06950      71 DELTX(I,1)=XMU*ABS(X(I,1))
06960         CALL FFUNC2(VTON,VTF1,X,U,FBAR0)
06970 C       PRINT 51,FBAR0
06980 C    51 FORMAT(*FBAR0=*,G15.4/3(6X,G15.4/))
06990         DO 68 J=1,4
07000 C       IF(J.NE.2) GO TO 54
07010 C       IF(MODE.EQ.2) GO TO 72
07020      54 X(J,1)=X(J,1)+DELTX(J,1)
07030         IT=0
07040         SC1=ZETA2(VTON,X,U)
07050      67 DZETA2=(ZETA2(VTON+DELTON,X,U)-SC1)/DELTON
07060         IT=IT+1
07070         VTON=VTON-SC1/DZETA2
07080         SC1=ZETA2(VTON,X,U)
```

```
07090        IF(ABS(SC1).LT.EPS) GO TO 64
07100        IF(IT.LT.NIT) GO TO 67
07110        PRINT 61,IT,SC1
07120     61 FORMAT(*MAX ITERATION ON TON. IT=*,I3,* SC1=*,E12.6/)
07130        GO TO 70
07140     64 IF(MODE.EQ.2) GO TO 81
07150        VTF1=TP-VTON
07160        IF(LDUTY.EQ.2) VTF1=TF
07170        GO TO 65
07180     81 CONTINUE
07190        CALL PHIM2(VTON,F1,PHI1)
07200        CALL DMAT2(VTON,F1,G1,PHI1,D1)
07210        CALL GTS2(Y,PHI1,X,D1,U)
07220        DELTX(2,1)=XMU*ABS(Y(2,1))
07230        IT=0
07240        B1=BMAT2(VTF1,Y,U)
07250     63 DB=(BMAT2(VTF1+DELTF1,Y,U)-B1)/DELTF1
07260        IT=IT+1
07270        VTF1=VTF1-B1/DB
07280        B1=BMAT2(VTF1,Y,U)
07290        IF(ABS(B1).LT.1.E-6*EPS) GO TO 65
07300        IF(IT.LT.NIT) GO TO 63
07310        PRINT 66,IT,B1
07320     66 FORMAT(*MAX ITERATION ON TF1. IT=*,I3,* SC1=*,E12.6/)
07330        GO TO 70
07340     65 CALL FFUNC2(VTON,VTF1,X,U,FBAR)
07350 C      PRINT 53,VTON,VTF1
07360 C   53 FORMAT(*VTON=*,G15.4/*VTF1=*,G15.4/)
07370        DO 69 I=1,4
07380     69 PSI(I,J)=(FBAR(I,1)-FBARO(I,1))/DELTX(J,1)
07390 C      PRINT 52,FBAR
07400 C   52 FORMAT(*FBAR=*,G15.4/3(5X,G15.4/))
07410        X(J,1)=X(J,1)-DELTX(J,1)
07420        GO TO 68
07430     72 DO 74 I=1,4
07440     74 PSI(I,2)=0.
07450     68 CONTINUE
07460     70 RETURN
07470        END
07480 ************************************************************
07490        SUBROUTINE FFUNC2(TON,TF1,X,U,F)
07500        DIMENSION TEMP1(4,4),TEMP2(4,4),PHI1(4,4),PHI2(4,4),
07510       XPHI3(4,4),D1(4,4),D2(4,4),D3(4,4),PHI(4,4),V(4,4),
07520       XFTEMP1(4,1),FTEMP2(4,1),F(4,1),X(4,1),U(4,1)
07530        COMMON/PARAM/F1(4,4),F2(4,4),F3(4,4),G1(4,4),G2(4,4),
07540       XG3(4,4)
07550        COMMON/EXTPAR/NIT,EPS,MTERMS,MIT,TP,TF,LDUTY,ET,MODE,XMU
07560        CALL PHIM2(TON,F1,PHI1)
07570        CALL PHIM2(TF1,F2,PHI2)
07580        CALL DMAT2(TON,F1,G1,PHI1,D1)
07590        CALL DMAT2(TF1,F2,G2,PHI2,D2)
```

```
      CALL RMMUL(PHI,PHI2,PHI1,4,4,4)
      CALL RMMUL(TEMP1,PHI2,D1,4,4,4)
      CALL RMADD(V,TEMP1,D2,4,4)
      IF(MODE.EQ.1) GO TO 15
      IF(LDUTY.EQ.1) GO TO 21
      TF2=TF-TF1
      GO TO 22
   21 CONTINUE
      TF2=TP-TF1-TON
   22 CONTINUE
      CALL PHIM2(TF2,F3,PHI3)
      CALL DMAT2(TF2,F3,G3,PHI3,D3)
      CALL RMMUL(TEMP2,PHI3,PHI,4,4,4)
      CALL RMCPY(PHI,TEMP2,4,4)
      CALL RMMUL(TEMP1,PHI3,V,4,4,4)
      CALL RMADD(V,TEMP1,D3,4,4)
   15 CONTINUE
      CALL STS2(F,PHI,X,V,U)
      IF(MODE.EQ.1)GO TO 2
      F(2,1)=0.
   2  RETURN
      END
*************************************************************
      SUBROUTINE GAMM2(GAM,TON,TF1,X,U)
      DIMENSION X(4,1),U(4,1),FBARO(4,1),FBAR(4,1),GAM(4,1),
     XPHI1(4,4),D1(4,4),TEMPY1(4,1),TEMPY2(4,1),Y(4,1)
      COMMON/PARAM/F1(4,4),F2(4,4),F3(4,4),G1(4,4),
     XG2(4,4),G3(4,4)
      COMMON/EXTPAR/NIT,EPS,NTERMS,MIT,TP,TF,LDUTY,ET,MODE,XMU
      VTON=TON
      VTF1=TF1
      DELTON=XMU*TON
      DELTF1=XMU*TF1
      DELU=XMU*ABS(U(1,1))
      CALL FFUNC2(VTON,VTF1,X,U,FBARO)
      U(1,1)=U(1,1)+DELU
      IT=0
      SC1=ZETA2(VTON,X,U)
   67 DZETA2=(ZETA2(VTON+DELTON,X,U)-SC1)/DELTON
      IT=IT+1
      VTON=VTON-SC1/DZETA2
      SC1=ZETA2(VTON,X,U)
      IF(ABS(SC1).LT.EPS) GO TO 64
      IF(IT.LT.NIT) GO TO 67
      PRINT 61,IT,SC1
   61 FORMAT(*MAX ITERATION ON TON. IT=*,I3,* SC1=*,E12.6/)
      GO TO 70
   64 CALL PHIM2(VTON,F1,PHI1)
      CALL DMAT2(VTON,F1,G1,PHI1,D1)
      CALL STS2(Y,PHI1,X,D1,U)
      IT=0
```

```
08110       IF(MODE.EQ.1) GO TO 65
08120       B1=BMAT2(VTF1,Y,U)
08130    63 DB=(BMAT2(VTF1+DELTF1,Y,U)-B1)/DELTF1
08140       IT=IT+1
08150       VTF1=VTF1-B1/DB
08160       B1=BMAT2(VTF1,Y,U)
08170       IF(ABS(B1).LT.1.E-6*EPS) GO TO 65
08180       IF(IT.LT.NIT) GO TO 63
08190       PRINT 66,IT,B1
08200    66 FORMAT(*MAX ITERATION ON TF1. IT=*,I3,* SC1=*,E12.6/)
08210       GO TO 70
08220    65 CALL FFUNC2(VTON,VTF1,X,U,FBAR)
08230       DO 69 I=1,4
08240    69 GAM(I,1)=(FBAR(I,1)-FBARO(I,1))/DELU
08250       U(1,1)=U(1,1)-DELU
08260    70 RETURN
08270       END
08280 ************************************************************
08290       SUBROUTINE XMAT2(TON,XMAT)
08300       DIMENSION PHI1(4,4),D1(4,4),PHI2(4,4),D2(4,4),
08310      XPHI3(4,4),D3(4,4)
08320       COMMON/PARAM/F1(4,4),F2(4,4),F3(4,4),G1(4,4),G2(4,4),
08330      XG3(4,4)
08340       COMMON/EXTPAR/NIT,EPS,NTERMS,MIT,TP,TF,LDUTY,ET,MODE,XMU
08350       COMMON/STATE/X(4,1),Y(4,1),Z(4,1),U(4,1)
08360       TF1=TP-TON
08370       IF(LDUTY.EQ.2) TF1=TF
08380       CALL STATE2(TON,TF1)
08390       CALL PHIM2(TF1,F2,PHI2)
08400       CALL DMAT2(TF1,F2,G2,PHI2,D2)
08410       XMAT=X(4,1)-PHI2(4,1)*Y(1,1)-PHI2(4,2)*Y(2,1)-Y(4,1)-
08420      X PHI2(4,3)*Y(3,1)-D2(4,2)*U(2,1)-D2(4,4)*U(4,1)-
08430      X D2(4,1)*U(1,1)
08440       RETURN
08450       END
08460 ************************************************************
08470       SUBROUTINE SMAT2(TON,TF1,SMAT,BMAT)
08480       DIMENSION PHI1(4,4),D1(4,4),
08490      XPHI2(4,4),D2(4,4),PHI3(4,4),D3(4,4)
08500       COMMON/PARAM/ F1(4,4),F2(4,4),F3(4,4),G1(4,4),G2(4,4),
08510      XG3(4,4)
08520       COMMON/EXTPAR/NIT,EPS,NTERMS,MIT,TP,TF,LDUTY,ET,MODE,XMU
08530       COMMON/STATE/X(4,1),Y(4,1),Z(4,1),U(4,1)
08540       PTF1=TF1
08550       DELTF1=XMU*TF1
08560       CALL STATE2(TON,TF1)
08570       IT=0
08580       B1=BMAT2(TF1,Y,U)
08590       IF(ABS(B1).LT.EPS) GO TO 32
08600    31 B2=BMAT2(TF1+DELTF1,Y,U)
08610       DBMAT2=(B2-B1)/DELTF1
```

```
08620       TF1=TF1-B1/DBMAT2
08630       B1=BMAT2(TF1,Y,U)
08640       IF(ABS(B1).LT.EPS) GO TO 32
08650       IT=IT+1
08660       IF(IT.LT.NIT) GO TO 31
08670       PRINT 33,IT,TF1,B1
08680    33 FORMAT(*MAX. ITERATION ON IT=*,I3,* TF1=*,G15.4,* B1=*,G15.4/)
08690       TF1=PTF1
08700    32 CONTINUE
08710       CALL STATE2(TON,TF1)
08720       CALL PHIM2(TF1,F2,PHI2)
08730       CALL DMAT2(TF1,F2,G2,PHI2,D2)
08740       CALL STS2(Z,PHI2,Y,D2,U)
08750       IF(LDUTY.EQ.1) GO TO 21
08760       TF2=TF-TF1
08770       GO TO 22
08780    21 CONTINUE
08790       TF2=TP-TON-TF1
08800    22 CONTINUE
08810       CALL PHIM2(TF2,F3,PHI3)
08820       CALL DMAT2(TF2,F3,G3,PHI3,D3)
08830       SMAT=X(4,1)-PHI3(4,1)*Z(1,1)-
08840      X PHI3(4,3)*Z(3,1)-Z(4,1)-D3(4,2)*U(2,1)
08850       BMAT=B1
08860    34 RETURN
08870       END
08880 ************************************************************
08890       FUNCTION BMAT2(TF1,Y,U)
08900       DIMENSION PHI2(4,4),D2(4,4),Y(4,1),U(4,1)
08910       COMMON/PARAM/ F1(4,4),F2(4,4),F3(4,4),G1(4,4),G2(4,4),
08920      XG3(4,4)
08930       COMMON/EXTPAR/NIT,EPS,NTERMS,MIT,TP,TF,LDUTY,ET,MODE,XMU
08940       CALL PHIM2(TF1,F2,PHI2)
08950       CALL DMAT2(TF1,F2,G2,PHI2,D2)
08960       BMAT2=PHI2(2,1)*Y(1,1)+PHI2(2,2)*Y(2,1)+
08970      Z D2(2,1)*U(1,1)+D2(2,4)*U(4,1)
08980       RETURN
08990       END

C
```

Appendix C. Buck-Boost Regulator Computer Program Description

The computer program is very similar to Appendix B Boost Regulator. Table B1 through B3 also apply in this program. Table C1 is example of nominal case data. Table C2 list the buck boost PAS computer program.

Table C1. BBPAS Program Parameters

RO	0.19
LO	2.2E-4
CO	6.0E-4
RS	0.013
RL	20.0
R1	56.258E3
R2	16.E3
R3	47.E3
R4	4.E4
R5	2.E3
C1	5.6E-9
C2	3.3E-8
TP	3.E-5
VO	28.
EISWIT	23.
XMU	0.01
EI	40.
ER	6.2
EQ	0.2
ED	0.7
ET	7.5
EPS	1.E-5
THETAO	0.
DELTHET	5.
THETAF	180.
N1T	10
N1	33
N3	22
EFF	.94
NTERMS	20
M1T	5

LRTL	0
NRL	2
DPRAM	0
PRAMF	0
LIST	1
LPEAK	0
LFE	0
NK	15
LFREQ	0
LPARAM	0
LCOMP	0
LPC	0
LSA	0
LRLPC	0
LCFR	0
LPEAK	0

NOTE: Resistance, inductance and capacitance are given on ohms, henries, and farads respectively. Voltages are given in volts. All other parameter units are identified in content.

APPENDIX C

TABLE C2. BUCK BOOST PAS COMPUTER PROGRAM

```
00100       PROGRAM PAS3(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,
00102      XTAPE3,TAPE4,TAPE7)
00110 ****** BUCK-BOOST PAS IMPLEMENT TRANSIENT HISTORY ALGORITHM
00120 ****** FOR VO COMPUTATION 4.4.78
00130 ****** BUCK-BOOST PAS MODIFY TF2 CUTOFF CRITERION
00140 ****** TF2.GE.TEPS=1X1P SPECIFIES MODE=2 OPERATION
00150 ****** MODIFY TRA AND LCA SEQUENCE IN ORDER TO UTILIZE
00160 ****** PROPER PSI MATRIX OPERATING POINT (EI/RL)
00170 ****** ABOVE CHANGES IMPLEMENTED 3.27.78
00180 ****** BUCK-BOOST PAS FOR TP CONST (LDUTY=1) AND
00190 ****** TF CONST (LDUTY=2) 3.23.78
00200 ****** TRA DELTA VO HISTORY 3.22.78
00210 ****** LCA LOAD CHANGE ANALYSIS RL TO RLSWIT 3.22.78
00212 ****** TAPE4 IS ASA DATA
00214 ****** TAPE3 IS TRA DATA
00220       DIMENSION RIPX(4,1),PSI(4,4),PSY(4,4),GAM(4,1),INT(8),
00230      XXFP(4,1),PRAM(10),H(4),R(4,5),ITBL(3),IVD(4),DELX(4,1),
00240      XPHI1(4,4),PHI2(4,4),PHI3(4,4),D1(4,4),D2(4,4),D3(4,4),
00250      XW(4,1),RPRAM(10),XTR(4,1),YTR(4,1),ZTR(4,1)
00260       COMMON /PARAM/F1(4,4),F2(4,4),F3(4,4),G1(4,4),G2(4,4),
00270      XG3(4,4)
00280       COMMON/EXTPAR/NIT,EPS,NTERMS,MIT,TP,TF,LDUTY,EI,MODE,XMU
00290       COMMON /STATE/X(4,1),Y(4,1),Z(4,1),U(4,1)
00300       EQUIVALENCE (PRAM(1),C1),(PRAM(2),C2),
00310      X(PRAM(3),R3),(PRAM(4),R4),(PRAM(5),R5),
00320      X(PRAM(6),RL),(PRAM(7),N3),(PRAM(8),LO),
00330      X(PRAM(9),CO),(PRAM(10),EI)
00340       REAL LO,N1,N3,KD,KL,KP,LP
00350       DATA R0,LO,CO,RS,RL/0.19,2.2E-4,6.E-4,0.013,40./
00360       DATA R1,R2,R3,R4,R5,C1,C2/56.258E3,16.E3,47.E3,4.E4,2.E3,
00370      X5.6E-9,3.3E-8/
00390       DATA EI,ER,EQ,ED,ET,EPS/40.,6.2,0.2,0.7,7.5,1.E-5/
00400       DATA TP,VO,EISWIT,XMU,KD/3.E-5,28.,40.,0.01,6.2/
00410       DATA THETA0,DELTHET,THETAF/0.,5.,180./
00430       DATA NIT,N1,N3,EFF/50,33.,22.,.94/
00440       DATA NTERMS,MIT,LRTL,NKL,DPRAM,PRAMF/20,5,0,10,4.,40./
00450       DATA LIST,LPEAK,LFE,NK,LFREQ/1,0,0,60,0/
00460       DATA LPARAM,LCOMP,LPC,LSA,LRLPC,LCFR,LPEAK/0,0,0,0,0,0,0/
00470       DATA RPRAM/2HC1,2HC2,2HR3,2HR4,2HR5,2HRL,2HN3,2HLO,
00480      X 2HCO,2HEI/
00490       DATA RLSWIT,LRL/600.,0./
00500       DATA TF,LDUTY,TPCON/1.514365E-5,1,3.E-5/
00510       DATA OLDTIME,LTR,LRESP,ISWIT,TFINAL/0.,1,0,2.E-3,13.9E-3/
00520       DATA MIR,SEPS/1,2.E-4/
00530       NAMELIST/PARAM/R0,LO,CO,RS,RL,R1,R2,R3,R4,R5,C1,C2,
00540      XN1,N3,EFF,EI,ER,EQ,ED,ET,TP,TF,VO
00550       NAMELIST/COMP/EPS,NIT,XMU,NTERMS,MIT
00560       NAMELIST/CONTRL/LPARAM,LCOMP,LPC,LSA,LRTL,LRLPC,LFREQ
00570      X,LCFR,LPEAK,LOAD,LDUTY,LRESP
```

```
00580       NAMELIST/RLPARAM/NRL,DPRAM,PRAM
00590       NAMELIST/FRPARAM/THETAO,DELTHET,THETAF
00600       NAMELIST/TAPARAM/E1,E1SWIT,NK
00610       NAMELIST/RPARAM/RL,RLSWIT,NK
00620       NAMELIST/TRPARAM/OLDTIME,TSWIT,IFINAL
00622       REWIND 4
00624       REWIND 3
00630       REWIND 7
00640 C     WRITE(6,99)
00650    99 FORMAT(1X,*PROGRAM FUNCTION*,13X,* CONTROL PARAMETER (1-YES 0-NO)*
00660      X /* CHANGE PARAM*,18X,*LPARAM*
00670      X /* CHANGE COMP*,19X,*LCOMP*
00680      X /* LIST PARAM, COMP*,14X,*LPC*
00690      X /* STABILITY ANALYSIS*,12X,*LSA*
00700      X /* ROOT LOCUS ANALYSIS*,11X,*LRTL*
00710      X /*       LIST PARAMETER CODE*,6X,*LRLPC*
00720      X /* AUDIO ANALYSIS*,16X,*LFREQ*
00730      X /*       CHANGE FREQ RANGE*,8X,*LCFR*
00740      X /* TRANSIENT ANALYSIS*,12X,*LPEAK*
00750      X /* TRANSIENT LOAD    *,12X,*LOAD*
00760      X /* DUTY CYCLE SCHEME *,12X,*LDUTY*
00770      X /* TRANSITION ALGORITHM *,6X,*LRESP*/)
00780   400 CONTINUE
00790       WRITE(6,401)
00800   401 FORMAT(1X,*ENTER N TO DISCONTINUE PAS, OTHERWISE Y*)
00810       READ(5,402) X1
00820   402 FORMAT(A1)
00830       IF(X1.EQ.1HN) STOP
00840       WRITE(6,403)
00850   403 FORMAT(1X,*INPUT PAS CONTROL PARAMETERS*)
00860       READ(5,CONTRL)
00870 C     WRITE(6,CONTRL)
00880       IF(LDUTY.EQ.1) GO TO 420
00890       WRITE(6,421)
00900   421 FORMAT(/1X,*CONSTANT TF DUTY CYCLE SCHEME*/)
00910       GO TO 422
00920   420 CONTINUE
00925       TP=TPCON
00930       WRITE(6,423)
00940   423 FORMAT(/1X,*CONSTANT TP DUTY CYCLE SCHEME*/)
00950   422 CONTINUE
00960       IF(LPARAM.EQ.0) GO TO 404
00970       READ(5,PARAM)
00980   404 IF(LCOMP.EQ.0) GO TO 405
00990       READ(5,COMP)
01000   405 IF(LPC.EQ.0) GO TO 406
01010       WRITE(6,PARAM)
01020       WRITE(6,COMP)
01030   406 IF(LSA.EQ.0) GO TO 407
01040       GO TO 5
01050   407 IF(LRTL.EQ.0) GO TO 408
```

```
01060       IF(LRLPC.EQ.0) GO TO 409
01070       WRITE(6,460)
01080   460 FORMAT(1X,*  CODE    PARAMETER*
01090      X /*   1=     C1*/*   2=     C2*/*   3=     R3*/
01100      X *   4=     R4*/*   5=     R5*/*   6=     RL*/
01110      X *   7=     N3*/*   8=     L0*/*   9=     C0*/
01120      X *  10=     E1*/)
01130   409 WRITE(6,410)
01140   410 FORMAT(1X,*INPUT ROOT LOCUS PARAMETERS*)
01150       READ(5,RLPARAM)
01160       WRITE(6,RLPARAM)
01170       GO TO 5
01180   408 IF(LFREQ.EQ.0) GO TO 411
01190       IF(LCFR.EQ.0) GO TO 5
01200       WRITE(6,413)
01210   413 FORMAT(1X,*INPUT FREQUENCY RANGE PARAMETERS*)
01220       READ(5,FRPARAM)
01230       WRITE(6,FRPARAM)
01240       GO TO 5
01250   411 IF(LPEAK.EQ.0) GO TO 415
01260       WRITE(6,414)
01270   414 FORMAT(1X,*INPUT TRANSIENT ANALYSIS PARAMETERS*)
01280       READ(5,TAPARAM)
01290       WRITE(6,TAPARAM)
01300       GO TO 5
01310   415 IF(LOAD.EQ.0) GO TO 424
01320       WRITE(6,416)
01330   416 FORMAT(1X,*INPUT LOAD CHANGE PARAMETERS*)
01340       READ(5,RPARAM)
01350       WRITE(6,RPARAM)
01360   424 CONTINUE
01370       IF(LRESP.EQ.0) GO TO 400
01380       WRITE(6,425)
01390   425 FORMAT(1X,*INPUT TRANSITION RESPONSE PARAMETERS*)
01400       READ(5,TRPARAM)
01410       WRITE(6,TRPARAM)
01420     5 CONTINUE
01430       IF(LRTL.EQ.0)GO TO 4
01440       WRITE(6,212)RPRAM(NRL),PRAM(NRL)
01450   212 FORMAT(//*ROOT LOCUS PARAMETER *,A2,* = *,G12.4)
01460     4 CONTINUE
01470       OLDEI=EI
01480       OLDRL=RL
01490       RL=RL/(RS+RL)
01500       KD=R2/(R1+R2)
01501       KP=N1/L0
01510       DO 11 I=1,4
01520       DO 11 J=1,4
01530       G1(I,J)=G2(I,J)=G3(I,J)=0.
01540    11 F1(I,J)=F2(I,J)=F3(I,J)=0.
01560    19 F1(1,1)=F2(1,1)=F3(1,1)=-1./(C0*(RL+RS))
```

```
01570         F1(2,2)=-R0/L0
01580         F1(3,3)=F2(3,3)=F3(3,3)=-1./(C2*R5)
01590         F1(3,1)=F2(3,1)=F3(3,1)=-KL*F1(3,3)
01600         F1(4,3)=F2(4,3)=F3(4,3)=1./(C1*R5)
01610         G1(4,2)=G2(4,2)=G3(4,2)=1./(C1*(R3+R1*KD))
01620         F1(4,1)=F3(4,1)=-KD*(KD*G1(4,2)+F1(4,3))
01630         G1(2,3)=G2(2,4)=-1/N1
01640         G1(2,1)=-G1(2,3)
01650         G1(4,3)=G2(4,4)=N3/(N1*C1*R4)
01660         G1(4,1)=-G1(4,3)
01670         F1(4,2)=N1*F1(2,2)*G1(4,1)
01680         F2(1,2)=KL*KP/C0
01690         F2(2,1)=-KL/N1
01700         F2(2,2)=KP*RS*F2(2,1)+F1(2,2)
01710         F2(3,2)=KP*RS*F2(3,1)
01720         F2(4,1)=KL*G1(4,3)+F1(4,1)
01730         F2(4,2)=KP*RS*F2(4,1)+F1(4,2)
01750 C       WRITE(6,93)((F1(I,J),J=1,4),I=1,4)
01760 C       WRITE(6,98)((G1(I,J),J=1,4),I=1,4)
01770 C       WRITE(6,97)((F2(I,J),J=1,4),I=1,4)
01780 C       WRITE(6,96)((G2(I,J),J=1,4),I=1,4)
01790 C       WRITE(6,95)((F3(I,J),J=1,4),I=1,4)
01800 C       WRITE(6,94)((G3(I,J),J=1,4),I=1,4)
01810 C    93 FORMAT(/*F1=*,/4(4G15.4/)/)
01820 C    98 FORMAT(/*G1=*,/4(4G15.4/)/)
01830 C    97 FORMAT(/*F2=*,/4(4G15.4/)/)
01840 C    96 FORMAT(/*G2=*,/4(4G15.4/)/)
01850 C    95 FORMAT(/*F3=*,/4(4G15.4/)/)
01860 C    94 FORMAT(/*G3=*/4(4G15.4/)/)
01870         U(1,1)=EI
01880         U(2,1)=ER
01890         U(3,1)=EQ
01900         U(4,1)=ED
01910         H(2)=H(3)=H(4)=0.
01920         H(1)=KL
01930         P0=V0**2/RL
01940         LP=L0*P0
01950         QA=EFF*EI*(EI-EQ)*(V0+ED-EQ)
01960         QB=-2.*LP*(V0+ED-EI)
01970         QC=QB*TF
01980       7 CONTINUE
01990         IF(LDUTY.EQ.1) GO TO 41
02000         DN=EFF*EI*(EI-EQ)
02010         TON=(LP+SQRT(LP**2+2.*DN*LP*TF))/DN
02020         TF1=TON*(EI-EQ)/(V0+ED)
02030         TF2=TF-TF1
02040         TP=TON+TF
02050         GO TO 42
02060      41 CONTINUE
02070         GTON=SQRT(2.*LP*TP/EFF)
02080         TON=GTON/SQRT(EI*(EI-EQ))
```

```
02090        TF1=TON*(EI-EQ)/(V0+ED)
02100        TF2=TP-TON-TF1
02110     42 CONTINUE
02120        TEPS=.01*TP
02130        DELTON=XMU*TON
02140        IF(TF2.GE.TEPS) GO TO 6
02150        MODE=1
02160        IF(LDUTY.EQ.1) GO TO 43
02170        TON=TF*(V0+ED)/(EI-EQ)
02180        TF1=TF
02190        TP=TON+TF1
02200        GO TO 44
02210     43 TON=TP*(V0+ED)/(V0+ED-EQ+EI)
02220        TF1=TP-TON
02230     44 CONTINUE
02240        OTON=TON
02250        OTF1=TF1
02260        CALL STATE2(TON,TF1)
02270 C      WRITE(6,27)TON,TF1,TF2,X,Y,Z
02280 C   27 FORMAT(*MODE=1*/*APPROXIMATE STEADY STATE*/*TON=*,G15.4,
02290 C     X *  TF1=*,G15.4,*  TF2=*,G15.4/*X=*,4G15.4/*Y=*,4G15.4/
02300 C     X *Z=*,4G15.4)
02310        DELTON=XMU*TON
02320        IT=0
02330        CALL XMAT2(TON,XC1)
02340     18 CALL XMAT2(TON+DELTON,XC2)
02350        DMATCH=(XC2-XC1)/DELTON
02360        TON=TON-XC1/DMATCH
02370        CALL XMAT2(TON,XC1)
02380        IT=IT+1
02390        IF(ABS(XC1).LE.EPS) GO TO 31
02400        IF(IT.LT.NIT) GO TO 18
02410        TON=OTON
02420        TF1=OTF1
02430        CALL STATE2(TON,TF1)
02440        WRITE(6,34)
02450     34 FORMAT(/*EXCEED MAX. ITERATION FOR THE EXACT STATE*/
02460       X*APPROXIMATE STATE IS CALCULATED*/)
02470     31 CONTINUE
02480        IF(LDUTY.EQ.1) GO TO 45
02490        TF1=TF
02500        TP=TON+TF1
02510        GO TO 46
02520     45 CONTINUE
02530        TF1=TP-TON
02540     46 CONTINUE
02550        CALL PHIM2(TF1,F2,PHI2)
02560        CALL DMAT2(TF1,F2,G2,PHI2,D2)
02570        ERR=X(4,1)-PHI2(4,1)*Y(1,1)-PHI2(4,2)*Y(2,1)-
02580       X PHI2(4,3)*Y(3,1)-Y(4,1)-
02590       X D2(4,2)*U(2,1)-D2(4,4)*U(4,1)
```

```
02600       DO 32 I=1,4
02610    32 RIPX(I,1)=X(I,1)-Y(I,1)
02620       CALL STATE2(TON,TF1)
02630       TPCT=100.*TON/TP
02640 C     IF(LRTL.EQ.2) GO TO 36
02650       IF(MTR.EQ.2) GO TO 36
02660       WRITE(6,35) EI,RL,LDUTY,MODE,TON,TF1,TP,TPCT,X,Y,RIPX,
02670      X ERR,IT,TEPS
02680    35 FORMAT(/*EI  =*,G12.4,*  RL=*,G12.4,*  LDUTY=*,I3,
02690      X *  MODE=*,I3/*TON =*,G12.4,*  TF1=*,G12.4,*  TP=*,G12.4,
02700      X *  TPCT=*,G12.4/*X   =*,4G12.4/*Y   =*,4G12.4/
02710      X *RIPX=*,4G12.4/*ERR =*,G12.4,*  IT=*,I3,*  TEPS=*,G12.4/)
02720    36 CONTINUE
02730       GO TO 56
02740     6 CONTINUE
02750       PTON=TON
02760       PTF1=TF1
02770       MODE=2
02780       CALL STATE2(TON,TF1)
02790 C     WRITE(6,17)TON,TF1,TF2,X,Y,Z
02800 C  17 FORMAT(*MODE=2*/*APPROXIMATE STEADY STATE*/*TON=*,G15.4,
02810 C     X*  TF1=*,G15.4,*  TF2=*,G15.4/
02820 C     X*X=*,4G15.4/*Y=*,4G15.4/*Z=*,4G15.4/)
02830       IT=0
02840       CALL SMAT2(TON,TF1,SC1,BC1)
02850     9 CALL SMAT2(TON+DELTON,TF1,SC2,BC1)
02860       DSMAT2=(SC2-SC1)/DELTON
02870       TON=TON-SC1/DSMAT2
02880       CALL SMAT2(TON,TF1,SC1,BC1)
02890       IT=IT+1
02900       IF(ABS(SC1).LE.SEPS) GO TO 10
02910       IF(IT.LT.NIT) GO TO 9
02920       TF1=PTF1
02930       TON=PTON
02940       CALL SMAT2(TON,TF1,SC1,BC1)
02950    10 CONTINUE
02960       IF(LDUTY.EQ.1) GO TO 47
02970       TF2=TF-TF1
02980       TP=TON+TF
02990       GO TO 48
03000    47 CONTINUE
03010       TF2=TP-TF1-TON
03020    48 CONTINUE
03030    13 DO 12 I=1,3
03040    12 RIPX(I,1)=X(I,1)-Y(I,1)
03050       CALL STATE2(TON,TF1)
03060       TPCT=100.*TON/TP
03070 C     IF(LRTL.EQ.2) GO TO 56
03080       IF(MTR.EQ.2) GO TO 36
03090       WRITE(6,55) EI,RL,LDUTY,MODE,TON,TF1,TF2,TP,X,Y,Z,
03100      X RIPX,SC1,IT,TPCT,BC1
```


C-2

```
03110     55 FORMAT(/*E1  =*,G12.4,*  RL=*,G12.4,*  LDUTY=*,I3,
03120       X *  MODE=*,I3,/*TON =*,G12.4,*  TF1=*,G12.4,*  TF2=*,G12.4,
03130       X *  TF=*,G12.4/*X   =*,4G12.4/*Y   =*,4G12.4/*Z   =*,
03140       X 4G12.4/*RIPX=*,4G12.4/*SC1 =*,G12.4,*  IT=*,I3,
03150       X *  TPCT=*,G12.4,*  RC1=*,G12.4/)
03160
03170     56 CONTINUE
03180        CALL PSIM2(PSI,TON,TF1,X,U)
03190        CALL RMCPY(PSY,PSI,4,4)
03200        ITBL(1)=4
03210        ITBL(3)=0
03220        CALL QRAL(PSY,R,4,M,V,INT,IVD,ITBL)
03230 C      IF(LRTL.NE.2) GO TO 67
03240 C      WRITE(6,68) MODE,TPCT
03250 C 67   CONTINUE
03260 C 68   FORMAT(*MODE =*,I2,* DUTY CYCLE =*,F5.2/)
03270 C      IF (LRTL.EQ.2) GO TO 72
03280        WRITE(6,70) ((PSI(I,J),J=1,4),I=1,4)
03290     70 FORMAT(*PSI=*,/4(4G15.4/))
03300        IF(MTR.EQ.2) GO TO 76
03310     72 WRITE(6,74)((R(I,J),J=1,2),I=1,4)
03320     74 FORMAT(5X,4HREAL,11X,4HIMAG,11X,4HREAL,11X,4HIMAG,
03330       X /2(4G15.4/))
03340    76  CONTINUE
03350        IF(LRESP.EQ.0) GO TO 550
03360        CALL PHIM2(TON,F1,PHI1)
03370        CALL DMAT2(TON,F1,G1,PHI1,D1)
03380        CALL PHIM2(TF1,F2,PHI2)
03390        CALL DMAT2(TF1,F2,G2,PHI2,D2)
03400        IF(MODE.EQ.1) GO TO 589
03410        CALL PHIM2(TF2,F3,PHI3)
03420        CALL DMAT2(TF2,F3,G3,PHI3,D3)
03430   589  CONTINUE
03440        TIME=OLDTIME
03450        IF(LTR.EQ.2) GO TO 501
03460        IF(MODE.EQ.1) GO TO 500
03470        CALL RMCPY(ZTR,Z,4,1)
03480        VOTR=RL*ZTR(1,1)-VO
03490 C      WRITE(6,599)TIME,ZTR,VOTR
03500        WRITE(7,599)TIME,ZTR,VOTR
03510   599  FORMAT(6G12.4)
03520        GO TO 501
03530   500  CONTINUE
03540        CALL RMCPY(YTR,Y,4,1)
03550        VOTR=RL*YTR(1,1)+RS*RL*YTR(2,1)-VO
03560 C      WRITE(6,599)TIME,YTR,VOTR
03570        WRITE(7,599)TIME,YTR,VOTR
03580   501  CONTINUE
03590        TIME=TIME+TON
03600        IF(MODE.EQ.1) GO TO 502
03610        CALL SIS2(XTR,PHI3,ZTR,D3,U)
```

```
03620       GO TO 503
03630  502  CONTINUE
03640       CALL STS2(XTR,PHI2,YTR,D2,U)
03650  503  CONTINUE
03660       VOTR=KL*XTR(1,1)-VO
03670 C     WRITE(6,599)TIME,XTR,VOTR
03680       WRITE(7,599)TIME,XTR,VOTR
03690       TIME=TIME+TF1
03700       CALL STS2(YTR,PHI1,XTR,D1,U)
03710       VOTR=KL*YTR(1,1)+RS*KL*YTR(2,1)-VO
03720 C     WRITE(6,599)TIME,YTR,VOTR
03730       WRITE(7,599)TIME,YTR,VOTR
03740       IF(MODE.EQ.1) GO TO 504
03750       TIME=TIME+TF2
03760       CALL STS2(ZTR,PHI2,YTR,D2,U)
03770       VOTR=KL*ZTR(1,1)-VO
03780 C     WRITE(6,599)TIME,ZTR,VOTR
03790       WRITE(7,599)TIME,ZTR,VOTR
03800  504  CONTINUE
03810       IF(TIME.GT.TFINAL) GO TO 505
03820       IF(TIME.LT.TSWIT) GO TO 501
03830       IF(LTR.EQ.2) GO TO 501
03840       IF(MODE.EQ.2) GO TO 506
03850       CALL RMCPY(ZTR,YTR,4,1)
03860  506  CONTINUE
03870       U(1,1)=EISWIT
03880       EI=EISWIT
03890       OLDTIME=TIME
03900       LTR=2
03910       GO TO 7
03920  505  CONTINUE
03930       OLDTIME=0.
03940       LTR=1
03950       MTR=2
03960       LRESP=0
03970       GO TO 7
03980  550  CONTINUE
03990       MTR=1
04000       IF(LFREQ.EQ.0) GO TO 150
04010       CALL GAMM2(GAM,TON,TF1,X,U)
04020       WRITE(6,75)  (GAM(I,1),I=1,4)
04030    75 FORMAT(/*GAM=*/4(G15.4/))
04040    69 CALL FREQ2(PSI,GAM,H,THETA0,THETAF,DELTHET,EI,VO)
04050       READ(5,FRPARAM)
04060       IF(THETAF.EQ.0.)GO TO 149
04070       WRITE(6,FRPARAM)
04080       GO TO 69
04090   149 CONTINUE
04100       LFREQ=0
04110  150  CONTINUE
04120       IF(LFE.EQ.1) GO TO 160
```

```
      IF(LPEAK.EQ.0) GO TO 200
      DO 152 I=1,4
152   XFP(I,1)=X(I,1)
      EI=EISWIT
      U(1,1)=EI
      LPE=1
      WRITE(6,189)
189   FORMAT(1X,*SET UP EI STEP INPUT*)
      GO TO 7
160   CALL OVSH2(PSI,XFP,NK)
200   CONTINUE
      LPEAK=LPE=0
      EI=OLDEI
      U(1,1)=EI
      IF(LRL.EQ.1) GO TO 161
      IF(LOAD.EQ.0) GO TO 201
      RL=RLSWIT
      LRL=1
      DO 153 I=1,4
153   XFP(I,1)=X(I,1)
      WRITE(6,188)
188   FORMAT(1X,*SET UP RL LOAD CHANGE*)
      GO TO 4
161   CALL OVSH2(PSI,XFP,NK)
201   CONTINUE
      LOAD=LRL=0
      RL=OLDRL
      IF(LRTL.EQ.0) GO TO 300
      PRAM(NRL)=PRAM(NRL)+DPRAM
      IF(PRAM(NRL).GT.PRAMF) GO TO 300
      LRTL=2
      GO TO 5
300   CONTINUE
      GO TO 400
      END
*******************************************************
      SUBROUTINE OVSH2(PSI,XFP,NK)
      DIMENSION PSI(4,4),XFP(4,1),DELXP(4,1),TVEC(4,1)
      COMMON/STATE/X(4,1),Y(4,1),Z(4,1),U(4,1)
      COMMON/EXTPAR/NIT,EPS,NTERMS,MIT,TP,TF,LDUTY,EI,MODE,XMU
      DATA RS,C0/.013,6.E-4/
      G=RS*C0/TP
      DO 10 I=1,4
10    DELXP(I,1)=XFP(I,1)-X(I,1)
      DELV0=DELXP(1,1)+G*DELXP(1,1)
      DO 30 NN=1,NK
      WRITE(6,12) NN,DELXP,DELV0
      WRITE(3,12) NN,DELXP,DELV0
12    FORMAT(I3,5G12.3)
      CALL RMMUL(TVEC,PSI,DELXP,4,4,1)
      DELV0=TVEC(1,1)+G*(TVEC(1,1)-DELXP(1,1))
```

```
04640       DO 14 I=1,4
04650    14 DELXP(I,1)=TVEC(I,1)
04660    30 CONTINUE
04670       RETURN
04680       END
04690 ************************************************************
04700       FUNCTION ZETA2(T,X,U)
04710       DIMENSION PHI1(4,4),D1(4,4),X(4,1),U(4,1)
04720       COMMON/PARAM/F1(4,4),F2(4,4),F3(4,4),G1(4,4),
04730      XG2(4,4),G3(4,4)
04740       COMMON/EXTPAR/NIT,EPS,NTERMS,MIT,TP,TF,LDUTY,ET,MODE,XMU
04750       CALL PHIM2(T,F1,PHI1)
04760       CALL DMAT2(T,F1,G1,PHI1,D1)
04770       ZETA2=-ET+PHI1(4,1)*X(1,1)+PHI1(4,2)*X(2,1)+
04780      XPHI1(4,3)*X(3,1)+X(4,1)+D1(4,1)*U(1,1)+
04790      XD1(4,2)*U(2,1)+D1(4,3)*U(3,1)
04800       RETURN
04810       END
04820 ************************************************************
04830       SUBROUTINE FREQ2(PSI,DVEC,H,THETAO,THETAF,DELTHET,EI,VO)
04840       DIMENSION PSI(4,4),DVEC(4,1),H(4)
04850       DIMENSION A(4,4),AINV(4,4),B(4,4),U(4,4),V(4,4),TEMP1(4,4)
04860       DIMENSION TEMP2(4,4),TVEC1(4,1),TVEC2(4,1)
04870       COMMON/EXTPAR/NIT,EPS,NTERMS,MIT,TP,TF,LDUTY,ET,MODE,XMU
04880       DEGRAD=180./3.1415927
04890       RADDEG=1./DEGRAD
04900       THETA=THETAO-DELTHET
04910       DO 2 I=1,4
04920       DO 2 J=1,4
04930     2 B(I,J)=0.
04940 C     WRITE(4,59)EI,MODE
04950    59 FORMAT(*AUDIOSUSCEPTIBILITY*,* EI=*,F6.2,* MODE=*,I2)
04960       WRITE(6,1) EI
04970     1 FORMAT(//*EI=*,E12.6/,
04980      X * THETA   FREQ (HZ)   DBEL*,5X,*G*,11X,*REG*,10X,*IMG*,
04990      X 6X,*PHASE*)
05000     5 CONTINUE
05010       THETA=THETA+DELTHET
05020       THET=RADDEG*THETA
05030       FRE=THET/(6.2831853*TP)
05040       RX=COS(THET)
05050       RY=SIN(THET)
05060       DO 12 I=1,4
05070       DO 10 J=1,4
05080    10 A(I,J)=-PSI(I,J)
05090       A(I,I)=A(I,I)+RX
05100    12 B(I,I)=RY
05110       IF(ABS(RY).LT.1.E-10) GO TO 25
05120       RYIV=1./RY
05130       CALL RAMUL(TEMP1,A,A,4,4,4)
05140       CALL RASCLR(TEMP2,RYIV,TEMP1,4,4)
```

```
05150       DO 14 I=1,4
05160    14 TEMP2(I,1)=TEMP2(I,1)+R1
05170       CALL RMINV(V,TEMP2,4)
05180       CALL RMMUL(U,A,V,4,4,4)
05190       DO 16 I=1,4
05200       DO 16 J=1,4
05210       U(I,J)=RYIV*U(I,J)
05220    16 V(I,J)=-V(I,J)
05230       GO TO 50
05240    25 CALL RMINV(AINV,A,4)
05250       DO 30 I=1,4
05260       DO 30 J=1,4
05270    30 TEMP1(I,J)=AINV(I,J)*RY**2
05280       CALL RMADD(TEMP2,A,TEMP1,4,4)
05290       CALL RMINV(U,TEMP2,4)
05300       CALL RMMUL(V,AINV,U,4,4,4)
05310       DO 34 I=1,4
05320       DO 34 J=1,4
05330    34 V(I,J)=-RY*V(I,J)
05340    50 CONTINUE
05350       CALL RMMUL(TVEC1,U,DVEC,4,4,1)
05360       CALL RMMUL(TVEC2,V,DVEC,4,4,1)
05370       GRE=GIM=0.
05380       DO 55 I=1,4
05390       GRE=GRE+H(I)*TVEC1(I,1)
05400    55 GIM=GIM+H(I)*TVEC2(I,1)
05410       G=SQRT(GRE**2+GIM**2)
05420       DBEL=20.*ALOG10(G*E1/V0)
05430       PHASE=DEGRAD*ATAN2(GIM,GRE)
05440       WRITE(6,60) THETA,FRE,DBEL,G,GRE,GIM,PHASE
05450       WRITE(4,58)FRE,DBEL,PHASE
05460    58 FORMAT(G15.4,2F12.4)
05470    60 FORMAT(F6.2,E12.4,F9.2,3E12.4,F9.2)
05480       IF(THETA.LT.THETAF-0.5*DELTHET) GO TO 5
05490   100 CONTINUE
05500       RETURN
05510       END
05520 ************************************************************
05530       SUBROUTINE PHIM2(T,F,PHI)
05540       DIMENSION P1(4,4),P(4,4),PHI(4,4),PA(4,4),P2(4,4),F(4,4)
05550       COMMON/EXTPAR/NIT,EPS,NTERMS,MIT,TP,TF,LDUTY,ET,MODE,XMU
05560       CALL RMSCLR(P1,T,F,4,4)
05570       CALL RMCPY(PHI,P1,4,4)
05580       DO 160 I=1,4
05590   160 PHI(I,I)=P1(I,I)+1.
05600       CALL RMCPY(P,P1,4,4)
05610       DO 131 M=1,NTERMS
05620       M=M+1
05630       A=1./M
05640       CALL RMSCLR(PA,A,P1,4,4)
05650       CALL RMMUL(P2,PA,P,4,4,4)
```

```
05660       CALL RMCPY(P,P2,4,4)
05670       CALL RMADD(PHI,PHI,P,4,4)
05680       DO 132 I=1,4
05690       DO 132 J=1,4
05700   132 ANORM=ANORM+ABS(P(I,J))
05710       IF (ANORM.LT.EPS) GO TO 133
05720   131 CONTINUE
05730   133 RETURN
05740       END
05750 ********************************************************
05760       SUBROUTINE DMAT2(T,F,G,PHI,D)
05770       DIMENSION TEMP1(4,4),TEMP2(4,4),WO(4,4),W2(4,4),WN(4,4),
05780      XW(4,4),F(4,4),G(4,4),B(4,4),PHI(4,4),D(4,4)
05790       COMMON/EXTPAR/NIT,EPS,NTERMS,MIT,TP,TF,LDUTY,ET,MODE,XMU
05800       DT=T/MIT
05810       DTK=T/MIT
05820       DT1=DT/2
05830       DO 136 I=1,4
05840       DO 136 J=1,4
05850   136 WO(I,J)=0.
05860       DO 137 I=1,4
05870   137 WO(I,I)=WO(I,I)+DT1
05880       CALL PHIM2(-T,F,W2)
05890       CALL RMSCLR(WN,DT1,W2,4,4)
05900       CALL RMADD(W,WO,WN,4,4)
05910       K=MIT-1
05920       DO 141 L=1,K
05930       CALL PHIM2(-DT,F,TEMP1)
05940       CALL RMSCLR(TEMP2,DTK,TEMP1,4,4)
05950       CALL RMADD(W,W,TEMP2,4,4)
05960       DT=DT+DTK
05970   141 CONTINUE
05980       CALL RMMUL(B,PHI,W,4,4,4)
05990       CALL RMMUL(D,B,G,4,4,4)
06000       RETURN
06010       END
06020 ********************************************************
06030       SUBROUTINE STATE2(TON,TF1)
06040       DIMENSION PHI1(4,4),PHI2(4,4),PHI3(4,4),D1(4,4),D2(4,4),
06050      XD3(4,4),TEMP1(4,4),TEMP2(4,4),PHI(4,4),V(4,4),VU(4,1)
06060       DIMENSION W(4,1),TV1(4,1),TV2(4,1)
06070       COMMON/PARAM/ F1(4,4),F2(4,4),F3(4,4),G1(4,4),G2(4,4),
06080      XG3(4,4)
06090       COMMON/EXTPAR/NIT,EPS,NTERMS,MIT,TP,TF,LDUTY,ET,MODE,XMU
06100       COMMON/STATE/X(4,1),Y(4,1),Z(4,1),U(4,1)
06110       CALL PHIM2(TON,F1,PHI1)
06120       CALL DMAT2(TON,F1,G1,PHI1,D1)
06130       CALL PHIM2(TF1,F2,PHI2)
06140       CALL DMAT2(TF1,F2,G2,PHI2,D2)
06150       CALL RMMUL(PHI,PHI2,PHI1,4,4,4)
06160       CALL RMMUL(TEMP1,PHI2,D1,4,4,4)
```

```
06170       CALL RMADD(V,TEMP1,D2,4,4)
06180       IF(MODE.EQ.1) GO TO 15
06190       IF(LDUTY.EQ.1) GO TO 31
06200       TF2=TP-TF1
06210       GO TO 32
06220    31 CONTINUE
06230       TF2=TP-TON-TF1
06240    32 CONTINUE
06250       CALL PHIM2(TF2,F3,PHI3)
06260       CALL DMAT2(TF2,F3,G3,PHI3,D3)
06270       CALL RMMUL(TEMP2,PHI3,PHI,4,4,4)
06280       CALL RMCPY(PHI,TEMP2,4,4)
06290       CALL RMMUL(TEMP1,PHI3,V,4,4,4)
06300       CALL RMADD(V,TEMP1,D3,4,4)
06310    15 CONTINUE
06320       CALL RMMUL(VU,V,U,4,4,1)
06330       DEN = (1.-PHI(1,1))*(1.-PHI(2,2)) - PHI(1,2)*PHI(2,1)
06340       DET = DEN * (1. - PHI(3,3))
06350       X(1,1)=((1.-PHI(2,2))*VU(1,1) + PHI(1,2)*VU(2,1))/DEN
06360       X(3,1)=((PHI(2,1)*PHI(3,2)+PHI(3,1)*(1.-PHI(2,2)))*VU(1,1)
06370      X +(PHI(3,2)*(1.-PHI(1,1))+PHI(1,2)*PHI(3,1))*VU(2,1))/DET
06380      X +VU(3,1)/(1.-PHI(3,3))
06390       IF(MODE.EQ.2) GO TO 25
06400       X(2,1)=(PHI(2,1)*VU(1,1)+(1.-PHI(1,1))*VU(2,1))/DEN
06410       GO TO 24
06420    25 X(2,1)=0.0
06430    24 CONTINUE
06440       X(4,1)=ET-PHI1(4,1)*X(1,1)-PHI1(4,2)*X(2,1)-
06450      ZPHI1(4,3)*X(3,1)-D1(4,1)*U(1,1)-D1(4,3)*U(3,1)-
06460      ZD1(4,2)*U(2,1)
06470       CALL STS2(W,PHI,X,V,U)
06480       CALL STS2(Y,PHI1,X,D1,U)
06490       IF(MODE.EQ.1) GO TO 23
06500       CALL STS2(Z,PHI2,Y,D2,U)
06510       Z(2,1)=0.
06520    23 CONTINUE
06530 C     WRITE(6,99)((PHI1(I,J),J=1,4),I=1,4)
06540 C     WRITE(6,98)((D1(I,J),J=1,4),I=1,4)
06550 C     WRITE(6,97)((PHI2(I,J),J=1,4),I=1,4)
06560 C     WRITE(6,96)((D2(I,J),J=1,4),I=1,4)
06570       IF(MODE.EQ.1)GO TO 35
06580 C     WRITE(6,89)((PHI3(I,J),J=1,4),I=1,4)
06590 C     WRITE(6,88)((D3(I,J),J=1,4),I=1,4)
06600    35 CONTINUE
06610 C     WRITE(6,95)((PHI(I,J),J=1,4),I=1,4)
06620 C     WRITE(6,94)((V(I,J),J=1,4),I=1,4)
06630 C 99  FORMAT(/*PHI1=*,/4(4G15.4/))
06640 C 98  FORMAT(/*D1=*,/4(4G15.4/))
06650 C 97  FORMAT(/*PHI2=*,/4(4G15.4/))
06660 C 96  FORMAT(/*D2=*,/4(4G15.4/))
06670 C 89  FORMAT(/*PHI3=*,/4(4G15.4/))
```

```
C 88  FORMAT(/*U3=*,/4(4G15.4/)/)
C 95  FORMAT(*PHI=*,/4(4G15.4/))
C 94  FORMAT(*V=*,/4(4G15.4/))
      RETURN
      END
*******************************************************
      SUBROUTINE STS2(U2,PHI,W1,D,U)
      DIMENSION PHI(4,4),W1(4,1),W2(4,1),D(4,4),U(4,1),
     XTEMPY1(4,1),TEMPY2(4,1)
      CALL RMMUL(TEMPY1,PHI,W1,4,4,1)
      CALL RMMUL(TEMPY2,D,U,4,4,1)
      CALL RMADD(W2,TEMPY1,TEMPY2,4,1)
      RETURN
      END
*******************************************************
      SUBROUTINE PSIM2(PSI,TON,TF1,X,U)
      DIMENSION X(4,1),FBAR0(4,1),FBAR(4,1),PSI(4,4),
     XU(4,1),Y(4,1),PHI1(4,4),D1(4,4),
     XTEMPY1(4,1),TEMPY2(4,1),DELTX(4,1)
      COMMON/PARAM/F1(4,4),F2(4,4),F3(4,4),G1(4,4),
     XG2(4,4),G3(4,4)
      COMMON/EXTPAR/NIT,EPS,NTERMS,MIT,TP,TF,LDUTY,ET,MODE,XMU
      VTON=TON
      VTF1=TF1
      DELTON=XMU*TON
      DELTF1=XMU*TF1
      DO 71 I=1,4
   71 DELTX(I,1)=XMU*ABS(X(I,1))
      CALL FFUNC2(VTON,VTF1,X,U,FBAR0)
C     PRINT 51,FBAR0
C  51 FORMAT(*FBAR0=*,G15.4/3(6X,G15.4/))
      DO 68 J=1,4
C     IF(J.NE.2) GO TO 54
C     IF(MODE.EQ.2) GO TO 72
   54 X(J,1)=X(J,1)+DELTX(J,1)
      IT=0
      SC1=ZETA2(VTON,X,U)
   67 DZETA2=(ZETA2(VTON+DELTON,X,U)-SC1)/DELTON
      IT=IT+1
      VTON=VTON-SC1/DZETA2
      SC1=ZETA2(VTON,X,U)
      IF(ABS(SC1).LT.EPS) GO TO 64
      IF(IT.LT.NIT) GO TO 67
      PRINT 61,IT,SC1
   61 FORMAT(*MAX ITERATION ON TON. IT=*,I3,* SC1=*,E12.6/)
      GO TO 70
   64 IF(MODE.EQ.2) GO TO 81
      VTF1=TP-VTON
      IF(LDUTY.EQ.2) VTF1=TF
      GO TO 65
   81 CONTINUE
```

```
07190       CALL PHIM2(VTON,F1,PHI1)
07200       CALL BMAT2(VTON,F1,G1,PHI1,B1)
07210       CALL SIS2(Y,PHI1,X,B1,U)
07220       DELTX(2,1)=XMU*ABS(Y(2,1))
07230       IT=0
07240       B1=BMAT2(VTF1,Y,U)
07250    63 DB=(BMAT2(VTF1+DELTF1,Y,U)-B1)/DELTF1
07260       IT=IT+1
07270       VTF1=VTF1-B1/DB
07280       B1=BMAT2(VTF1,Y,U)
07290       IF(ABS(B1).LT.1.E-6*EPS) GO TO 65
07300       IF(IT.LT.MIT) GO TO 63
07310       PRINT 66,IT,B1
07320    66 FORMAT(*MAX ITERATION ON TF1. IT=*,I3,* SC1=*,E12.6/)
07330       GO TO 70
07340    65 CALL FFUNC2(VTON,VTF1,X,U,FBAR)
07350 C     PRINT 53,VTON,VTF1
07360 C  53 FORMAT(*VTON=*,G15.4/*VTF1=*,G15.4/)
07370       DO 69 I=1,4
07380    69 PSI(I,J)=(FBAR(I,1)-FBAR0(I,1))/DELTX(J,1)
07390 C     PRINT 52,FBAR
07400 C  52 FORMAT(*FBAR=*,G15.4/3(5X,G15.4/))
07410       X(J,1)=X(J,1)-DELTX(J,1)
07420       GO TO 68
07430    72 DO 74 I=1,4
07440    74 PSI(1,2)=0.
07450    68 CONTINUE
07460    70 RETURN
07470       END
07480 ************************************************************
07490       SUBROUTINE FFUNC2(TON,TF1,X,U,F)
07500       DIMENSION TEMP1(4,4),TEMP2(4,4),PHI1(4,4),PHI2(4,4),
07510      XPHI3(4,4),B1(4,4),B2(4,4),B3(4,4),PHI(4,4),V(4,4),
07520      XFTEMP1(4,1),FTEMP2(4,1),F(4,1),X(4,1),U(4,1)
07530       COMMON/PARAM/F1(4,4),F2(4,4),F3(4,4),G1(4,4),G2(4,4),
07540      XG3(4,4)
07550       COMMON/EXTPAR/MIT,EPS,NTERMS,MIT,TP,TF,LDUTY,ET,MODE,XMU
07560       CALL PHIM2(TON,F1,PHI1)
07570       CALL PHIM2(TF1,F2,PHI2)
07580       CALL BMAT2(TON,F1,G1,PHI1,B1)
07590       CALL BMAT2(TF1,F2,G2,PHI2,B2)
07600       CALL RMMUL(PHI,PHI2,PHI1,4,4,4)
07610       CALL RMMUL(TEMP1,PHI2,B1,4,4,4)
07620       CALL RMADD(V,TEMP1,B2,4,4)
07630       IF(MODE.EQ.1) GO TO 15
07640       IF(LDUTY.EQ.1) GO TO 21
07650       TF2=TF-TF1
07660       GO TO 22
07670    21 CONTINUE
07680       TF2=TP-TF1-TON
07690    22 CONTINUE
```

```
      CALL PHIM2(TF2,F3,PHI3)
      CALL DMAT2(TF2,F3,G3,PHI3,D3)
      CALL RMMUL(TEMP2,PHI3,PHI,4,4,4)
      CALL RMCPY(PHI,TEMP2,4,4)
      CALL RMMUL(TEMP1,PHI3,V,4,4,4)
      CALL RMADD(V,TEMP1,D3,4,4)
   15 CONTINUE
      CALL STS2(F,PHI,X,V,U)
      IF(MODE.EQ.1)GO TO 2
      F(2,1)=0.
    2 RETURN
      END
*****************************************************
      SUBROUTINE GAMM2(GAM,TON,TF1,X,U)
      DIMENSION X(4,1),U(4,1),FBARO(4,1),FBAR(4,1),GAM(4,1),
     XPHI1(4,4),D1(4,4),TEMPY1(4,1),TEMPY2(4,1),Y(4,1)
      COMMON/PARAM/F1(4,4),F2(4,4),F3(4,4),G1(4,4),
     XG2(4,4),G3(4,4)
      COMMON/EXTPAR/NIT,EPS,NTERMS,MIT,TP,TF,LDUTY,ET,MODE,XMU
      VTON=TON
      VTF1=TF1
      DELTON=XMU*TON
      DELTF1=XMU*TF1
      DELU=XMU*ABS(U(1,1))
      CALL FFUNC2(VTON,VTF1,X,U,FBARO)
      U(1,1)=U(1,1)+DELU
      IT=0
      SC1=ZETA2(VTON,X,U)
   67 DZETA2=(ZETA2(VTON+DELTON,X,U)-SC1)/DELTON
      IT=IT+1
      VTON=VTON-SC1/DZETA2
      SC1=ZETA2(VTON,X,U)
      IF(ABS(SC1).LT.EPS) GO TO 64
      IF(IT.LT.NIT) GO TO 67
      PRINT 61,IT,SC1
   61 FORMAT(*MAX ITERATION ON TON. IT=*,I3,* SC1=*,E12.6/)
      GO TO 70
   64 CALL PHIM2(VTON,F1,PHI1)
      CALL DMAT2(VTON,F1,G1,PHI1,D1)
      CALL STS2(Y,PHI1,X,D1,U)
      IT=0
      IF(MODE.EQ.1) GO TO 65
      B1=BMAT2(VTF1,Y,U)
   63 DB=(BMAT2(VTF1+DELTF1,Y,U)-B1)/DELTF1
      IT=IT+1
      VTF1=VTF1-B1/DB
      B1=BMAT2(VTF1,Y,U)
      IF(ABS(B1).LT.1.E-6*EPS) GO TO 65
      IF(IT.LT.NIT) GO TO 63
      PRINT 66,IT,B1
   66 FORMAT(*MAX ITERATION ON TF1. IT=*,I3,* SC1=*,E12.6/)
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      GO TO 70
   65 CALL FFUNC2(VTON,VTF1,X,U,FBAR)
      DO 69 I=1,4
   69 GAM(I,1)=(FBAR(I,1)-FBAR0(I,1))/DELU
      U(1,1)=U(1,1)-DELU
   70 RETURN
      END
C*******************************************************
      SUBROUTINE XMAT2(TON,XMAT)
      DIMENSION PHI1(4,4),D1(4,4),PHI2(4,4),D2(4,4),
     XPHI3(4,4),D3(4,4)
      COMMON/PARAM/F1(4,4),F2(4,4),F3(4,4),G1(4,4),G2(4,4),
     XG3(4,4)
      COMMON/EXTPAR/NIT,EPS,NTERMS,MIT,TP,TF,LDUTY,ET,MODE,XMU
      COMMON/STATE/X(4,1),Y(4,1),Z(4,1),U(4,1)
      TF1=TP-TON
      IF(LDUTY.EQ.2) TF1=TF
      CALL STATE2(TON,TF1)
      CALL PHIM2(TF1,F2,PHI2)
      CALL DMAT2(TF1,F2,G2,PHI2,D2)
      XMAT=X(4,1)-PHI2(4,1)*Y(1,1)-PHI2(4,2)*Y(2,1)-Y(4,1)-
     X PHI2(4,3)*Y(3,1)-D2(4,2)*U(2,1)-D2(4,4)*U(4,1)
      RETURN
      END
C*******************************************************
      SUBROUTINE SMAT2(TON,TF1,SMAT,BMAT)
      DIMENSION PHI1(4,4),D1(4,4),
     XPHI2(4,4),D2(4,4),PHI3(4,4),D3(4,4)
      COMMON/PARAM/ F1(4,4),F2(4,4),F3(4,4),G1(4,4),G2(4,4),
     XG3(4,4)
      COMMON/EXTPAR/NIT,EPS,NTERMS,MIT,TP,TF,LDUTY,ET,MODE,XMU
      COMMON/STATE/X(4,1),Y(4,1),Z(4,1),U(4,1)
      PTF1=TF1
      DELTF1=XMU*TF1
      CALL STATE2(TON,TF1)
      IT=0
      B1=BMAT2(TF1,Y,U)
      IF(ABS(B1).LT.EPS) GO TO 32
   31 B2=BMAT2(TF1+DELTF1,Y,U)
      DBMAT2=(B2-B1)/DELTF1
      TF1=TF1-B1/DBMAT2
      B1=BMAT2(TF1,Y,U)
      IF(ABS(B1).LT.EPS) GO TO 32
      IT=IT+1
      IF(IT.LT.MIT) GO TO 31
      PRINT 33,IT,TF1,B1
   33 FORMAT(*MAX. ITERATION ON IT=*,I3,* TF1=*,G15.4,* B1=*,G15.4/)
      TF1=PTF1
   32 CONTINUE
      CALL STATE2(TON,TF1)
      CALL PHIM2(TF1,F2,PHI2)
```

```
08730        CALL BMAT2(TF1,F2,G2,PHI2,D2)
08740        CALL STS2(Z,PHI2,Y,D2,U)
08750        IF(LDUTY.EQ.1) GO TO 21
08760        TF2=TF-TF1
08770        GO TO 22
08780     21 CONTINUE
08790        TF2=TP-TON-TF1
08800     22 CONTINUE
08810        CALL PHIM2(TF2,F3,PHI3)
08820        CALL BMAT2(TF2,F3,G3,PHI3,D3)
08830        SMAT=X(4,1)-PHI3(4,1)*Z(1,1)-
08840       X PHI3(4,3)*Z(3,1)-Z(4,1)-D3(4,2)*U(2,1)
08850        BMAT=B1
08860     34 RETURN
08870        END
08880 ************************************************************
08890        FUNCTION BMAT2(TF1,Y,U)
08900        DIMENSION PHI2(4,4),D2(4,4),Y(4,1),U(4,1)
08910        COMMON/PARAM/ F1(4,4),F2(4,4),F3(4,4),G1(4,4),G2(4,4),
08920       XG3(4,4)
08930        COMMON/EXIPAR/NIT,EPS,NTERMS,MIT,TP,TF,LDUTY,EI,MODE,XMU
08940        CALL PHIM2(TF1,F2,PHI2)
08950        CALL BMAT2(TF1,F2,G2,PHI2,D2)
08960        BMAT2=PHI2(2,1)*Y(1,1)+PHI2(2,2)*Y(2,1)+
08970       Z D2(2,4)*U(4,1)
08980        RETURN
08990        END

C
```

APPENDIX D: Derivations of Constraints for Boost and Buck-Boost Converters

D.1 Basic relationships for Boost converter

a. Input-output voltage relationship

Transistor on: $\Delta i_{L5}(+)$ = current increment in L_5

$$= \frac{E_i\, T_{on}}{L_5}$$

Transistor off: $\Delta i_{L5}(-)$ = current decrement in L_5

$$= \frac{(E_0 - E_i)T_{off}}{L_5}$$

When in steady state, $\Delta i_{L5}(+) = \Delta i_{L5}(-)$

so $$\frac{E_i T_{on}}{L_5} = \frac{(E_0 - E_i)T_{off}}{L_5}$$

that is $$E_0 = E_i \frac{T}{T_{off}} \qquad (D.1)$$

b. Peak to peak ripple current through energy storage inductor

$$\Delta i_{L5}(+) = \frac{E_i T_{on}}{L_5} \qquad (D.2)$$

By substituting T_{on} from (A.1)

into the above equation for $\Delta i_{L5}(+)$

we can get $$\Delta i_{L5}(+) = \frac{E_i(E_0 - E_i)}{L_5 E_0 F} = 2d \qquad (D.3)$$

D.2 Derivations of transistor switching loss

a. Saturation loss

Examining the current waveform through transistor of Fig. 2.2

$$\text{During } T_{on}: \quad i_Q = \frac{2d}{T_{on}} t + (I_i - d) \tag{D.4}$$

$$\text{Saturation loss} = \frac{1}{T} \int_0^{T_{on}} V_{st}\, i_Q(t)dt \tag{D.5}$$

$$= \frac{T_{on}}{T} I_i V_{st}$$

From (A.1) we can get

$$T_{on} = \frac{E_O - E_i}{E_O} T$$

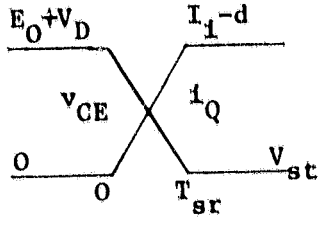
Also the average input current $I_i = \frac{P_O}{\text{eff } E_i}$

substitute I_i and T_{on} into the above equation, we can get

$$\text{saturation loss} = \frac{P_O V_{st} (E_O - E_i)}{\text{eff } E_i E_O} \tag{D.6}$$

b. Transistor turn-on loss

Switching waveform during turn on:



$$v_{CE}(t) = -\frac{E_O + V_D - V_{st}}{T_{sr}} t + (E_O + V_D) \tag{D.7}$$

$$i_Q(t) = \frac{I_i - d}{T_{sr}} t \tag{D.8}$$

$$\text{Average turn-on power loss} = \frac{1}{T}\int_0^{T_{sr}} v_{CE}(t)i_Q(t)dt$$

$$= \frac{1}{T}\int_0^{T_{sr}}\left[-\frac{E_0+V_D-V_{st}}{T_{sr}}t + E_0+V_D\right]\left[\frac{I_1-d}{T_{sr}}t\right]dt$$

$$= \frac{T_{sr}F(E_0+V_D+2V_{st})}{6}\left[\frac{P_0}{\text{eff } E_1} - \frac{E_1(E_0-E_1)}{2L_5\ E_0F}\right] \qquad \text{(D.9)}$$

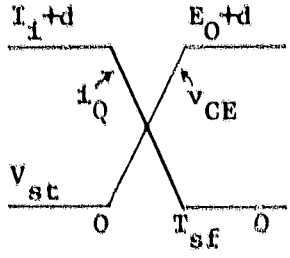
where $I_1 = \dfrac{P_0}{\text{eff } E_1}$

d = Half of the peak-peak ripple through L_5

$$= \frac{E_1(E_0-E_1)}{2L_5\ E_0\ F} \qquad \text{(D.10)}$$

c. Transistor turn-off loss

Switching waveform during turn-off



$$i_Q = \frac{(I_1+d)(t-T_{sf})}{-T_{sf}} \qquad \text{(D.11)}$$

$$v_{CE} = \frac{E_0+d-V_{st}}{T_{sf}}t + V_{st} \qquad \text{(D.12)}$$

Note that transistor current at the moment of turn-off is I_1+d instead of I_1-d.

$$\text{The turn-off power loss} = \frac{1}{T}\int_0^{T_{sf}} v_{CE}(t)i_C(t)dt$$

$$= \frac{1}{6}(E_O + V_D + 2V_{st})\left[\frac{P_O}{\text{eff}E_1} + \frac{E_1(E_O-E_1)}{2L_5E_OF}\right]T_{sf}F \qquad (D.13)$$

where $I_1 = \dfrac{P_O}{\text{eff } E_1}$

$$d = \frac{E_1(E_O-E_1)}{2L_5E_OF}$$

D.3 Derivation of diode switching loss

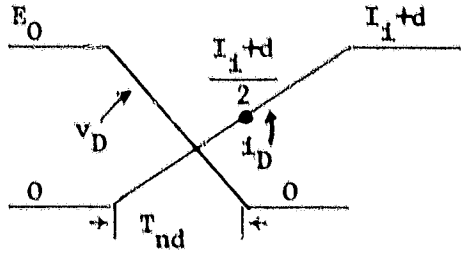
a. Diode conduction loss

Examining the i_D waveform as shown in Fig. 2.2

Diode conduction loss

$$= \frac{I_1V_D(T - T_{on})}{T} = I_1V_D\left(1 - \frac{T_{on}}{T}\right) = \frac{P_OV_D}{\text{eff } E_O} \qquad (D.14)$$

b. Diode turn-on loss



$$i_D = \frac{I_1+d}{2T_{nd}}\,t \qquad (D.15)$$

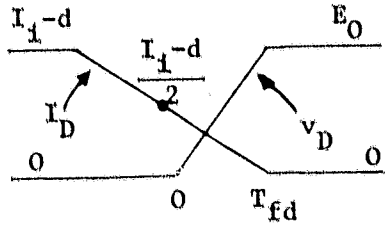
$$v_D = \frac{-E_O}{T_{nd}}\,(t - T_{nd}) \qquad (D.16)$$

$$\text{turn-on loss} = \frac{1}{T}\int_0^{T_{nd}} i_D(t)v_D(t)dt \qquad (D.17)$$

$$= \frac{1}{T}\int_0^{T_{nd}} \frac{-(I_i+d)E_O}{2\,T_{nd}^2}(t - T_{nd})t\,dt$$

$$= \frac{E_O T_{nd} F}{12}\left[\frac{P_O}{effE_i} + \frac{E_i(E_O-E_i)}{2L_5\,E_O F}\right]$$

c. Diode turn-off loss



$$i_D = -\frac{I_i-d}{2T_{fd}}(t - T_{fd}) \qquad (D.18)$$

$$v_D = \frac{E_O}{T_{fd}}t \qquad (D.19$$

$$\text{turn-off loss} = \frac{1}{T}\int_0^{T_{fd}} i_D(t)v_D(t)dt$$

$$= \frac{1}{T}\int_0^{T_{fd}} -\frac{(I_i - d)E_O}{2T_{fd}^2}(t^2 - T_{fd}t)dt$$

$$= \frac{E_O T_{fd} F}{12}\left[\frac{E_O}{effE_i} - \frac{E_i(E_O - E_i)}{2L_5\,E_O F}\right] \qquad (D.20)$$

D.4 Output filter ESR loss

Examining the i_{C_6} waveform of Fig. 2.2

During T_{ON}: $i_{C_6} = -I_O$ (D.21)

During T_{OFF}: $i_{C_6} = \frac{2d}{T_{on} - T}(t - T_{on}) + (I_i - I_O + d)$ (D.22)

$$P_{cap} = \text{ESR loss} = \frac{1}{T}\int_0^T i_{C_6}^2 R_6 \, dt$$

$$= \left[\frac{1}{T}\int_0^{T_{on}} i_{C_6}^2(t)dt + \frac{1}{T}\int_{T_{on}}^{T} i_{C_6}^2(t)dt\right] R_6 \quad \text{(D.23)}$$

By substituting (A.21) and (A.22) into the above integration formula, we can get

$$P_{cap} = R_6\,[(i_{C_6})\text{rms}]^2 = \left\{(1 - \frac{E_i}{E_O})(\frac{P_O}{E_O})^2 + \frac{E_i}{E_O}\left[\frac{E_i^2(E_O - E_i)^2}{12L_5^2E_O^2F^2} + (\frac{P_O}{\text{eff } E_i} - \frac{P_O}{E_O})^2\right]\right\}R_6 \quad \text{(D.24)}$$

D.5 Parasitic resistances for L_1, L_2, L_5

R (winding resistance) $= \rho\frac{\ell}{A}$ (D.25)

where $A = A_C$ (copper cross section area)

By definition of winding pitch factor F_C

$$F_C = \frac{\text{mean length per turn of winding}}{\text{core circumference}}$$

$$F_C = \frac{\ell_1}{4\sqrt{A_1}} \text{ (Toroidal core with square cross-section area)}$$

$$R_1 = \rho \frac{4F_C\sqrt{A_1}N_1}{A_{C1}} , \tag{D.26}$$

$$\text{so } R_1A_{C1} - 4\rho F_C\sqrt{A_1}N_1 = 0 \tag{D.27}$$

The equation for L_2, L_5 can be likewise derived

$$R_2A_{C2} - 4\rho F_C\sqrt{A_2}N_2 = 0 \tag{D.28}$$

$$R_5A_{C5} - 4\rho F_C\sqrt{A_5}N_5 = 0 \tag{D.29}$$

D.6 Operating flux density constraint

Maximum flux linkage of $L_5 = L_5(\Delta I)_{max} = N_5(B_{S5}A_5)$

$$\text{where } (\Delta I)_{max} = I_i + d = \frac{P_O}{\text{eff } E_i} + \frac{E_i(E_O - E_i)}{2L_5\ E_OF} \tag{D.30}$$

$$\text{so } \frac{P_O}{\text{eff } E_i} + \frac{E_i(E_O - E_i)}{2L_5\ E_OF} = N_5B_{S5}A_5$$

$$\text{that is } N_5A_5 = \frac{L_5}{B_{S5}}\left[\frac{P_O}{\text{eff } E_i} + \frac{E_i(E_O - E_i)}{2L_5E_OF}\right] \tag{D.31}$$

Because L_1, L_2 do not handle AC ripple current

$$\text{so } N_1A_1 = \frac{L_1P_O}{\text{eff } E_iB_{S1}} \tag{D.32}$$

$$N_2A_2 = \frac{L_2P_O}{\text{eff } E_iB_{S2}} \tag{D.33}$$

D.7 Window area constraint

By definition of window fill factor

$$F_W = \frac{\text{core window area actually occupied by the core}}{\text{available core window area} = \Pi r^2}$$

$$F_W = \frac{N_1A_{C1}}{\Pi\left(\frac{Z_1}{2\Pi} - \frac{\sqrt{A_1}}{2}\right)^2} \tag{D.34}$$

where Z_1 = mean magnetic path length

A_1 = core cross section area

$\frac{Z_1}{2\Pi} - \frac{\sqrt{A_1}}{2}$ = radius of window area

so $$\left(\frac{N_1 A_{C1}}{\Pi F_W}\right)^{0.5} = \frac{Z_1}{2\Pi} - \frac{\sqrt{A_1}}{2} \qquad (D.35)$$

likewise, $$\left(\frac{N_2 A_{C2}}{\Pi F_W}\right)^{0.5} = \frac{Z_2}{2\Pi} - \frac{\sqrt{A_1}}{2} \qquad (D.36)$$

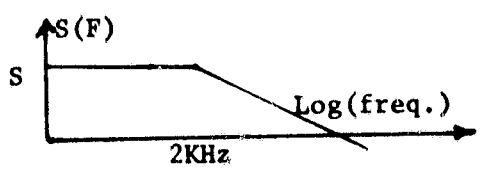
$$\left(\frac{N_5 A_{C5}}{\Pi F_W}\right)^{0.5} = \frac{Z_5}{2\Pi} - \frac{\sqrt{A_5}}{2} \qquad (D.37)$$

D.8 Frequency dependent source EMI constraint

Required attenuation at switching frequency

$$= \frac{\text{EMI requirement}}{\text{Fundamental switching current}}$$

where EMI requirement $= \frac{S}{\sqrt{1 + (\frac{F}{2000})^2}}$ (D.38)



Derivation of fundamental switching current component of i_{L5}

By examining Fig. 2.2

During T_{ON}: $i_{L5} = \frac{2d}{T_{on}} t + (I_i - d)$ (D.39)

During T_{OFF}: $i_{L5} = \frac{-2d}{T-T_{on}} (t - T) + (I_i - d)$ (D.40)

where $d = \dfrac{E_i(E_O - E_i)}{2L_5 E_O F}$

$$I_i = \frac{P_O}{\text{eff } E_i}$$

By Fourier series expansion:

$$i_{L5} = C_O + \sum_{k=1}^{\infty} A_k \cos k\omega_O t + \sum_{k=1}^{\infty} B_k \cos k\omega_O t \qquad \text{(D.41)}$$

Only the first harmonics are needed in EMI calculation, the higher order terms can be neglected.

$$A_1 = \frac{2}{T}\int_0^{T_{on}} i_{L5}(t) \cos \omega_O t \, dt + \frac{2}{T}\int_{T_{on}}^{T} i_{L5}(t) \sin \omega_O t \, dt \qquad \text{(D.42)}$$

$$B_1 = \frac{2}{T}\int_0^{T_{on}} i_{L5}(t) \sin \omega_O t \, dt + \frac{2}{T}\int_{T_{on}}^{T} i_{L5}(t) \sin \omega_O t \, dt \qquad \text{(D.43)}$$

Substitute (A.39) and (A.40) into the above equation and integrate, one can get

$$A_1 = \frac{E_O}{2\Pi^2 L_5 F} \cos\left[\frac{2\Pi(E_O - E_i)}{E_O}\right] - \frac{E_O}{2\Pi^2 L_5 F} = \frac{-E_O}{\Pi^2 L_5 F}\left[\sin\frac{\Pi(E_O - E_i)}{E_i}\right]^2 \qquad \text{(D.44)}$$

$$B_1 = 0$$

D.9 Output Ripple Voltage Constraints

The peak-to-peak output-voltage ripple is caused by two components: the capacitive component due to the ampere-second processed by C and the resistive component due to the ESR R_6 of C. It can be shown that the constraint concerning the sum of these two components can be expressed as:

$$V_{RIP}=(I_i + d)\ R_6 + \frac{P_O D}{2E_O C_6 F} \tag{D.45}$$

where $D = \frac{T_{on}}{T} = \frac{E_O - E_i}{E_O}$

Substituting $I_i = \frac{P_O}{eff\ E_i}$ and (A.3) into (A.45)

$$V_{RIP}=\left(\frac{P_O}{eff\ E_i} + \frac{E_i(E_O - E_i)}{2L_5 E_O F}\right)R_6 + \frac{P_O(E_O - E_i)}{2E_O{}^2 C_6 F} \tag{D.46}$$

$$V_R \triangleq \frac{V_{RIP}}{E_O}$$

$$= \left[\frac{P_O}{eff E_i E_O} + \frac{E_i(E_O - E_i)}{2L_5 E_O{}^2 F}\right]R_6 + \frac{P_O(E_O - E_i)}{2E_O{}^3 C_6 F} \tag{D.47}$$

APPENDIX E: Boost Converter Computer List

In this appendix an complete computer list for the Boost Converter is given for the user's reference. Before the user implements his own problem in the computer code, he must decide the design unknown variables. For the Boost Converter the variable array is shown in the following:

$x(1) = \sqrt{A_1}$ *

$x(2) = \sqrt{N_1}$

$x(3) = \sqrt{A_{c1}}$

$x(4) = \sqrt{A_2}$

$x(5) = \sqrt{N_2}$

$x(6) = \sqrt{A_{c2}}$

$x(7) = L_1$

$x(8) = eff$

$x(9) = C_3$

$x(10) = C_4$

$x(11) = Z_1$

$x(12) = Z_2$

$x(13) = R_1$

$x(14) = R_2$

$x(15) = R_3$

$x(16) = R_5$

$x(17) = \sqrt{A_5}$

$x(18) = \sqrt{N_5}$

$x(19) = \sqrt{A_{c5}}$

$x(20) = L_5$

$x(21) = C_6$

$x(22) = Z_5$

*In the process of optimization some variables take on negative values. In order to avoid the square root of negative numbers, certain variables are presented in this form in the program.

After the variable array is chosen, the user can start to simplify the constraints and choose the constant terms of the constraints, and then take the first derivatives of all the constraints and objective function with respect to the variables. Then he can put all this information in computer code ready to use in the program. The constant terms, constraint equations and their first derivatives are shown as follows:

E.1 Constant Terms

```
PI=3.141592654
XM1=4.*FC*DC
XM2=4.*RO*FC
XM3=1./(2.*PI)
XM4=PO/EI
XM5=SQRT(1./FW)
XM6=SQRT(1./(PI*FW))
XM7=PO/EO
XM8=(EO-EI)/EO
XM9=EI*(EO-EI)/(2.*EO*FR)
XM10=FR*(EO+VO+2.*VST)/6.
XM11=TSF+TSR
XM12=TSF-TSR
XM13=TNO+TFO+3.*TRE
XM14=TNC-TFO-3.*TRE
XM15=EO*FR/12.
XM16=EI/EO
XM17=80.*EI*.0022*SQRT(FR)
XM18=1.+1./PE2
XM19=FR/PE2
XM20=1./EO
XM21=1./EI
XM22=(1.+FR*FR/2000**2)*(EO*SIN(PI*XM8/XM16)/PI**2)**2
XM23=(S*FR)**2
XM24=S/(SQRT(1.+FR*FR/2000.**2)*EO/PI**2*SIN(PI*(EO-EI)/EI))
XM25=FR*FR*XM24
   XM26=XM24*FR
   XM27=4.0*PI*PI*FR*FR
   XM28=2.0*PI*FR/PE2
RCKCK=RCK*CK
```

E.2 Objective function and constraint equation

```
1000  Y1=DI*(X1*X1*X11+X4*X4*X12+X17*X17*X22)
      Y2=XM1*(X3*X3*X2*X2*X1+X6*X6*X5*X5*X4+X19*X19*X18*X18*X17)
      Y3=DK3*X9+DK4*X10+DK6*X21
      Y4=PO*(1./X8-1.)/KH+PO/X8/KS
      F=Y1+Y2+Y3+Y4
        F=F/FSCAL
      PIF=(XM4/X8)**2*(X13+X14)
      PQ=(XM4*XM8/X8)*(VST+.1*VBE)+XM10*(XM4*XM11/X8+XM9*XM12/X20)
      PD=XM7*VD/X8+XM15*(XM4*XM13/X8+XM9*XM14/X20)
      POF=(XM4/X8)**2*X16+XM9*XM9*(X16+RCKCK*XM16/X21)/(3.*X20*X20)+(
     1RCKCK/X21)*(XM16*XM7**2+XM7*XM4/X8/X8-2.*XM7*XM7/X8)
     2+XM17*XM8*X22/X18/X18
7       C(1)=PO*(1./X8-1.)-PIF-PQ-PD-POF
8       C(2)=X13*X3*X3-XM2*X1*X2*X2
12      C(3)=X14*X6*X6-XM2*X5*X5*X4
9       C(4)=X9*X9+X15*X15*X9**3/X7-PE1**2*(X10*X10+(X15*X15*X9/X7)*
     1(X9-X10*XM18)**2)
        C(5)=X1*X1*X2*X2-X7*XM4/(X8*BS1)
13      C(6)=X4*X4*X5*X5-X7*XM4/(X8*BS2*PE2)
        C(7)=XM6*X3*X2-X11*XM3+.5*X1
        C(8)=XM6*X6*X5-X12*XM3+.5*X4
2       C(9)=X17*X17*X19*X19-(X20/BS5)*(XM4/X8+XM9/X20)
        C(10)=XM6*X19*X19-X22*XM3+.5*X17
        C(11)  =  VR-(XM4/X8  +  XM9/X20)*RCKCK/X21/EO-XM7*XM8/2/FR/X21/EO
        C(12)=X16*X19*X19-XM2*X18*X18*X17
        C(13)=-1.0+XM26*XM27*X7*X10*X20*(XM28*X7/X15-1.0)
        C(14)=.969-X8
11      C(15)=RT-X13-X14
        C(16)=X10-1.E-6
        C(17)=X9  -  1.E-6
```

E.3 First derivatives of the constraints and the objective function

(1) Derivatives of objective function

```
      G(1)=DI*X11*2*X1+XM1*X3*X3*X2*X2
      G(2)=XM1*X3*X3*X1*2*X2
      G(3)=XM1*X2*X2*X1*2*X3
      G(4)=DI*X12*2*X4+XM1*X6*X6*X5*X5
      G(5)=XM1*X6*X6*X4*2*X5
      G(6)=XM1*X5*X5*X4*2*X6
      G(7)=0.0
      G(8)=-PO/X8/X8/KH-PO/KS/X8/X8
      G(9)=DK3
      G(10)=DK4
      G(11)=DI*X1*X1
      G(12)=DI*X4*X4
      G(17)=DI*X22*2*X17+XM1*X19*X19*X18*X18
      G(18)=XM1*X19*X19*X17*2*X18
      G(19)=XM1*X18*X18*X17*2*X19
      G(21)=DK6
      G(22)=DI*X17*X17
```

where G(i) is the derivatives of the objective function with respect to variables x(i)

ORIGINAL PAGE IS
OF POOR QUALITY

(2) Derivatives of the constraints

In the following, GC(j,i) means the derivatives of the constraint C(i) with respect to variable x(j).

```
      GC(o,1)=-PO/X8/X8+(2./X8**3)*XM4*XM4*(X1**X14)+(XM4*XM8/X8**2)*(V
     1ST+.1*VBE)+XM10*(XM4*XM11/X8**2)
     2+XM7*VD/X8**2+XM15*(XM4*XM13/X8**2)+(2./X8**3)*XM4**2*X16
     3+2.*RCKCK*XM7*XM4/X8**3/X21-2.*RCKCK*XM7**2/X8/X8/X21
      GC(13,1)=-(XM4/X8)**2
      GC(14,1)=-(XM4/X8)**2
      GC(16,1)=-(XM4/X8)**2-XM9**2/(3.*X20**2)
      GC(18,1)=2*XM17*XM8*X22/X18/X18/X18
      GC(20,1)=XM10*XM9*XM12/X20**2+XM15*XM9*XM14/X20**2
     1+2.*XM9**2*(X15+RCKCK*XM16/X21)/(3.*X20**3)
      GC(21,1)=XM9**2*XM16*RCKCK/X21/X21/(3.*X20**2)+(RCKCK/X21/X21)*
     1(XM16*XM7**2+XM7*XM4/X8/X8-2.*XM7**2/X8)
      GC(22,1)=-XM17*XM8/X18/X18
      GC(13,2)=X3*X3
      GC(3,2)=X13*2*X3
      GC(2,2)=-XM2*X1*2*X2
      GC(1,2)=-XM2*X2*X2
      GC(14,3)=X6*X6
      GC(6,3)=X14*2*X6
      GC(5,3)=-XM2*X4*2*X5
      GC(4,3)=-XM2*X5*X5
      GC(15,4)=2.*X15*X9**3/X7-PE1*PE1*(2.*X15*X9/X7)*(X9-X10*XM18)**2
      GC(9,4)=2.*X9+3.*X9*X9*X15*X15/X7-PE1*PE1*((X15*X15/X7)*(X9-X10*
     1XM18)**2+(X15*X15*X9/X7)*2.*(X9-X10*XM18))
      GC(7,4)=-X15*X15*X9**3/X7/X7+PE1*PE1*(X15*X15*X9/X7/X7)*(X9-X10*
     1XM18)**2
      GC(10,4)=-PE1*PE1*(2.*X10-(X15*X15*X9/X7)*2.*(X9-X10*XM18)*XM18)
      GC(1,5)=X2*X2*2*X1
      GC(2,5)=X1*X1*2*X2
      GC(7,5)=-XM4/(X8*BS1)
      GC(8,5)=X7*XM4/(BS1*X8*X8)
      GC(4,6)=X5*X5*2*X4
      GC(5,6)=X4*X4*2*X5
      GC(7,6)=-XM4/(X8*BS2*PE2)
      GC(8,6)=X7*XM4/(PE2*BS2*X8*X8)
      GC(2,7)=XM6*X3
      GC(3,7)=XM6*X2
      GC(11,7)=-XM3
      GC(1,7)=.5
      GC(5,8)=XM6*X6
      GC(6,8)=XM6*X5
      GC(12,8)=-XM3
      GC(4,8)=.5
      GC(17,9)=X18*X18*2*X17
      GC(18,9)=X17*X17*2*X18
      GC(20,9)=-(XM4/X8+XM9/X20)/BS5+(X20/BS5)*XM9/X20**2
      GC(8,9)=X20/BS5*XM4/X8**2
      GC(18,10)=XM6*X19
      GC(19,10)=XM6*X18
      GC(22,10)=-XM3
      GC(17,10)=.5
      GC(8,11) = (XM4/X8/X8)*RCKCK/X21/EC
      GC(20,11) = (XM9/X20/X20)*RCKCK/X21/EC
      GC(21,11)=(XM4/X8+XM9/X20)*RCKCK/X21/X21/EC+XM7*XM8/2/FR/X21/X21/E
     1C
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
GC(16,12)=X19*X19
GC(19,12)=X16*2*X19
GC(18,12)=-XM2*X17*2*X18
GC(17,12)=-XM2*X18*X18
GC(7,13)=2.0*XM26*XM27*XM28*X7*X10*X20/X15-XM26*XM27*X10*X20
 GC(10,13)=XM26*XM27*X7*X20*(XM28*X7/X15-1.0)
GC(15,13)=-XM26*XM27*XM28*X7**2*X10*X20/X15**2
 GC(20,13)=XM26*XM27*X7*X10*(XM28*X7/X15-1.0)
GC(8,14)=-1.
GC(13,15)=-1.
GC(14,15)=-1.
GC(10,16)=1.
GC(9,17)=1.
```

With the above information available, the user is ready to combine it with all the other subroutines and ready to run his own program. The detailed computer program is provided in the following. The main program is put in the front and the user's supplied subroutine ALAGB is put at the end.

E.4 Computer List

A complete listing of the boost converter optimization program is attached.

INPUT DATA

```
   N   ,   M   ,   K   , MAXFIN, IPR1, IPR2,  IW  ,    MODE
    22     39     12     600      1     6C1  2500       1
0.000025  (AKMIN)
 &CON
  PO=70.,EI=28.,EC=37.5,FC=1.9,FW=0.4,RO=1.724E-8,VST=0.25,
  VSE=0.8,TSR=0.195E-6,TSF=0.25E-6,VD=0.9,TNO=0.03E-6,TFD=0.05E-6,
  TRE=0.03E-6,PE1=2.0,PE2=3.,BS1=0.40,BS2=0.40,BSE=0.40,VR=0.01,
  RCK=0.03,CK=100.E-6,C1=7300.,DC=8900.,DK3=210.,DK4=1100.,DK6=72.,
  KS=30.8,KH=15.4,S=0.1,FR=60000.0,RT=0.1,&END
 &XIN
  X=0.0019389,5.5511,0.0003328,0.001518,4.0913,0.0005377,0.00001757,
  0.94815,0.00002167,0.00001085,0.01645,0.0171,0.07069,0.01152,
  0.06,0.01637,0.005276,2.7175,0.0007639,0.0000315,0.00005172,0.03249
 &END
```

```
C       THIS PROG PLACES CONSTRAINT ON C3 AND C4.
        REAL    L1,L2,L5,KC,DK3,DK4,DK6,KS,KH,S
        DIMENSION X(22)
        DIMENSION EPS(22)
        COMMON/CONS/XM1,XM2,XM3,XM4,XM5,XM6,XM7,XM8,XM9,XM10,XM11,          SUM00030
       1XM12,XM13,XM14,XM15,XM16,XM17,XM18,XM19,XM20,XM21,XM22,XM23,XM24,  SUM00035
       1XM25,XM26,XM27,XM28,PO,EI,EO,FC,FW,RO,VST,VBE,TSR,TSF,
       1  VC,TNC,TFC,TRE,PE1,PE2,B  S1,BS2,BS5,VR,RCKCK,DI,                SUM0011
       2  DC,DK3,DK4,DK6,KS,KH,S,FR,RT                                      SUM00
        NAMELIST/CON/PO,EI,EO,FC,FW,RO,VST,VBE,TSR,TSF,                    SUM00110
       1  VC,TNC,TFC,TRE,PE1,PE2,B  S1,BS2,BS5,VR,RCK,CK,DI,               SUM0011
       2  DC,DK3,DK4,DK6,KS,KH,S,FR,RT                                      SUM00
        NAMELIST/PARMET/ PIF,PC,PD,POF,PCAP,PMAG,PT,SIGMAP,WS,WP,
       2WI,WTW,WC,WMAG,W
        NAMELIST/PARMS/N,M,K,MAXFN,IPR1,IPR2,IW,MODE,AKMIN,DFN
        NAMELIST/XIN/X
        COMMON/ALAGE/C(150)
        COMMON/ALAGF/GC(25,50)                                             MAIN0050
        COMMON/ALAGG/T(150)                                                MAIN0060
        COMMON/ALAGI/G2P(325)                                              MAIN0070
        COMMON/SCALE/VSCAL(22),CSCAL(40),FSCAL
        READ(5,10)N,M,K,MAXFN,IPR1,IPR2,IW,MODE                           MAIN0110
   10   FORMAT(1615)                                                       MAIN0120
        READ(5,20)AKMIN,DFN                                                MAIN0130
        WRITE(6,PARMS)
        READ(5,CON)                                                        SUM00145
        WRITE(6,CON)                                                       SUM00150
        FSCAL=1.0
        CSCAL( 1)=1.0E+1
        CSCAL( 2)=1.0E-10
        CSCAL( 3)=1.0E-12
        CSCAL( 4)=1.0E-12
        CSCAL( 5)=1.0E-6
        CSCAL( 6)=1.0E-6
        CSCAL( 7)=1.0E-5
        CSCAL( 8)=1.0E-6
        CSCAL( 9)=1.0E-4
        CSCAL(10)=1.0E-4
        CSCAL(11)=1.0E-3
        CSCAL(12)=1.0E-11
        CSCAL(13)=1.0E-2
        CSCAL(14)=1.0E+0
        CSCAL(15)=1.0E+0
        CSCAL(16)=1.0E-5
        CSCAL(17)=1.0E-5
        DO 103 I=1,N
  103   CSCAL(I+17)=1.0
        VSCAL( 1)=1.0E-3
        VSCAL( 2)=1.0E 0
        VSCAL( 3)=1.0E-4
        VSCAL( 4)=1.0E-3
        VSCAL( 5)=1.0E 0
        VSCAL( 6)=1.0E-4
        VSCAL( 7)=1.0E-5
        VSCAL( 8)=1.0E-1
```

USER SUPPLIED MAIN PROGRAM

CONSTRAINT SCALING FACTORS

VARIABLE SCALING FACTORS

ORIGINAL PAGE IS OF POOR QUALITY

ORIGINAL PAGE IS
OF POOR QUALITY

```
      VSCAL( 9)=1.0E-6
      VSCAL(10)=1.0E-6
      VSCAL(11)=1.0E-2
      VSCAL(12)=1.0E-2
      VSCAL(13)=1.0E-2
      VSCAL(14)=1.0E-2
      VSCAL(15)=1.0E-2
      VSCAL(16)=1.0E-2
      VSCAL(17)=1.0E-3
      VSCAL(18)=1.0
      VSCAL(19)=1.0E-4
      VSCAL(20)=1.0E-5
      VSCAL(21)=1.0E-5
      VSCAL(22)=1.0E-2
      READ(5,XIN)                                          SUM00185
        WRITE(6,XIN)
      DO   111   I=1,N
111   EPS(I)=0.0001
C      EPS(1)=0.00001
C      EPS(2)=0.001
C        EPS(3)=0.000001
C      EPS(4)=0.00001
C      EPS(5)=0.001
C      EPS(6)=0.000001
C      EPS(7)=0.0000001
C      EPS(8)=0.0001
C      EPS(9)=0.00000001
C      EPS(10)=0.00000001
C      EPS(11)=0.0001
C      EPS(12)=0.0001
C      EPS(13)=0.0001
C      EPS(14)=0.00001
C        EPS(15)=0.0001
C        EPS(16)=0.00001
C      EPS(17)=0.00001
C      EPS(18)=0.001
C      EPS(19)=0.00001
C      EPS(20)=0.00000001
C      EPS(21)=0.00001
C        EPS(22)=0.0001
   20 FORMAT(8F10.2)                                       MAIN0160
      WRITE(6,6016) EPS(1)
 8016 FORMAT(F12.8 //)
      PI=3.141592654                                       SUM00315
      XM1=4.*FC*DC
      XM2=4.*RO*FC
      XM3=1./(2.*PI)
      XM4=PO/EI
      XM5=SQRT(1./FW)
      XM6=SQRT(1./(PI*FW))
      XM7=PO/EC
      XM8=(EC-EI)/EO
      XM9=EI*(EC-EI)/(2.*EO*FR)
      XM10=FR*(EC+VO+2.*VST)/6.
      XM11=TSF+TSR
```

CONSTANT TERMS

```
      XM12=TSF-TSR
      XM13=TRD+TFD+3.*TRE
      XM14=TRD-TFD-3.*TRE
      XM15=EC*FR/12.
      XM16=EI/EC
      XM17=30.*EI*.0022*SQRT(FR)
      XM18=1.+1./PE2
      XM19=FR/PE2
      XM20=1./EC
      XM21=1./EI
      XM22=(1.+FR*FR/2000**2)*(EC*SIN(PI*XM8/XM16)/PI**2)**2
      XM23=(S*FR)**2
      XM24=S/(SQRT(1.+FR*FR/2000.**2)*EC/PI**2*SIN(PI*(EC-EI)/EI))
      XM25=FR*FR*XM24
        XM26=XM24*FR
        XM27=4.0*PI*PI*FR*FR
        XM28=2.0*PI*FR/PE2
      RCKCK=RCK*CK
C     THIS CHECKS FOR SCALING
      X1  =X(1)
      X2  =X(2)
      X3  =X(3)
      X4  =X(4)
      X5  =X(5)
      X6  =X(6)
      X7  =X(7)
      X8  =X(8)
      X9  =X(9)
      X10  =X(10)
      X11  =X(11)
      X12  =X(12)
      X13  =X(13)
      X14  =X(14)
      X15  =X(15)
      X16  =X(16)
      X17  =X(17)
      X18  =X(18)
      X19  =X(19)
      X20  =X(20)
      X21  =X(21)
      X22  =X(22)
 1000 Y1=DI*(X1*X1*X11+X4*X4*X12+X17*X17*X22)
      Y2=XM1*(X3*X3*X2*X2*X1+X6*X6*X5*X5*X4+X19*X19*X18*X18*X17)
      Y3=CK3*X9+CK4*X10+CK6*X21
      Y4=PO*(1./X8-1.)/KH+PC/X8/KS
      F=Y1+Y2+Y3+Y4
      PIF=(XM4/X8)**2*(X13+X14)
      PC=(XM4*XM8/X3)*(VST+.1*VBE)+XM10*(XM4*XM11/X8+XM9*XM12/X20)
      PD=XM7*VD/X6+XM15*(XM4*XM13/X8+XM9*XM14/X20)
      PCF=(XM4/X8)**2*X16+XM9*XM9*(X16+RCKCK*XM16/X21)/(3.*X20*X20)+(
     1RCKCK/X21)*(XM16*XM7**2+XM7*XM4/X8/X8-2.*XM7*XM7/X8)
     2+XM17*XM9*X22/X18/X18
    7   C(1)=PC*(1./X8-1.)-PIF-PC-PC-PCF
    8   C(2)=X13*X3*X3-XM2*X1*X2*X2
   12   C(3)=X14*X6*X6-XM2*X5*X5*X4
```


ORIGINAL PAGE IS
OF POOR QUALITY

```
 9      C(4)=X9*X9+X15*X15*X9**3/X7-PE1**2*(X10*X10+(X15*X15*X9/X7)*
     1(X9-X10*XM18)**2)
       C(5)=X1*X1*X2*X2-X7*XM4/(X8*BS1)
 13    C(6)=X4*X4*X5*X5-X7*XM4/(X8*BS2*PE2)
       C(7)=XM6*X3*X2-X11*XM3+.5*X1
       C(8)=XM6*X6*X5-X12*XM3+.5*X4
 2     C(9)=X17*X17*X18*X18-(X20/BS5)*(XM4/X8+XM9/X20)
       C(10)=XM6*X19*X18-X22*XM3+.5*X17
       C(11) = VR-(XM4/X8 + XM9/X20)*RCKCK/X21/E0-XM7*XM8/2/FR/X21/E0
       C(12)=X16*X19*X19-XM2*X18*X18*X17
       C(13)=-1.0+XM26*XM27*X7*X10*X20*(XM28*X7/X15-1.0)
       C(14)=.969-X8
 11    C(15)=RT-X13-X14
       C(16)=X10-1.E-6
       C(17)=X9 -.1.E-6
            WRITE(6,1388)F
1388          FORMAT(F13.6)
            WRITE(6,10000)(C(I),I=1,17)
10000         FORMAT(4X,5(E14.7,5X))
            DO 101 I=1,N
            X(I)=X(I)/VSCAL(I)          <-- VARIABLE SCALING
101           CONTINUE
       WRITE(6,XIN)                                                  SUM00805
       CALL ALAGA(N,M,K,X,EPS,AKMIN,DFN,MAXFN,IPR1,IPR2,IW,MODE)     MAIN0170
       DO 102 I =1,N
102    X(I)=X(I)*VSCAL(I)          <-- SCALE THE VARIABLE BACK
            WRITE(6,XIN)
       XL2=X(7)/PE2
       X1  =X(1)
       X2  =X(2)
       X3  =X(3)
       X4  =X(4)
       X5  =X(5)
       X6  =X(6)
       X7  =X(7)
       X8  =X(8)
       X9  =X(9)
       X10  =X(10)
       X11  =X(11)
       X12  =X(12)
       X13  =X(13)
       X14  =X(14)
       X15  =X(15)
       X16  =X(16)
       X17  =X(17)
       X18  =X(18)
       X19  =X(19)
       X20  =X(20)
       X21  =X(21)
       X22  =X(22)
       TX1=X(1)*X(1)
       TX2=X(2)*X(2)
       TX3=X(3)*X(3)
       TX4=X(4)*X(4)
       TX5=X(5)*X(5)
```

```
      TX6=X(6)*X(6)
      TX7=X(17)*X(17)
      TX8=X(18)*X(18)
      TX9=X(19)*X(19)
      WRITE(6,9025)TX1,TX2,TX3,TX4                                          SUM00925
 9025 FORMAT('0A1=',G13.5,' N1=',G13.5,' AC1=',G13.5,' A2=',G13.5).         SUM00930
      WRITE(6,9026)TX5,TX6,X(7),XL2
 9026 FORMAT('0N2=',G13.5,' AC2=',G13.5,' L1=',G13.5,' L2=',G13.5)          SUM00940
      WRITE(6,9027)X(9),X(10),X(11),X(12)                                   SUM00945
 9027 FORMAT('0C3=',G13.5,' C4=',G13.5,' Z1=',G13.5,' Z2=',G13.5)           SUM00950
      WRITE(6,9028)X(13),X(14),X(15),X(16)                                  SUM00955
 9028 FORMAT('0R1=',G13.5,' R2=',G13.5,' R3=',G13.5,' R5=',G13.5)           SUM00960
      WRITE(6,9001)TX7,TX8,TX9,X(20)                                        SUM00965
 9001 FORMAT('0A5=',G13.5,' N5=',G13.5,' AC5=',G13.5,' L5=',G13.5)          SUM00970
      WRITE(6,9002)X(21),X(22)                                              SUM00975
 9002 FORMAT('0C6=',G13.5,' Z5=',G13.5)                                     SUM00980
      WRITE(6,9003)FR,X(8)
 9003 FORMAT('0FR=',G13.5,' EFF=',G13.5)
      PIF=(XM4/X8)**2*(X13+X14)
      PC=(XM4*XM9/X8)*(VST+.1*VEE)+XM10*(XM4*XM11/X8+XM9*XM12/X20)
      PD=XM7*VD/X8+XM15*(XM4*XM13/X8+XM9*XM14/X20)
      POF=(XM4/X8)**2*X16+XM5*XM9*(X16+RCKCK*XM16/X21)/(3.*X20*X20)+(
     2XM17*XM9*X22/X19/X18)
      PCAP=(RCKCK/X21)*(XM16*XM7**2+XM7*XM4/X8/X8-2.*XM7*XM7/X8)
      PMAG=PIF+POF
       PT=PIF+PC+PD+POF+PCAP
       SIGMAP=PC/X8-PD
      RATIO=XL2/X(7)
      WS=PC/X(8)/KS                                                         SUM01085
      WH=PC*(1./X(9)-1.)/KH                                                 SUM01090
      WI=DI*(X(1)*X(1)*X(11)+X(4)*X(4)*X(12)+X(17)*X(17)*X(22))
      WTW=XM1*(X(3)*X(3)*X(2)*X(2)*X(1)+X(6)*X(6)*X(5)*X(5)*X(4)+
     1 X(19)*X(19)*X(18)*X(16)*X(17))
      WC=CK3*X(9)+CK4*X(10)+CK6*X(21)
       WMAG=WI+WTW
      W=WI+WTW+WC+WS+WH                                                     SUM01115
       WRITE(6,XIN)
      WRITE(6,PARMET)
      WRITE(6,30)(X(I),I=1,N)                                               MAIN0180
   30 FORMAT('1',10X,'SOLUTION VECTOR',/,('0',10X,4E16.7))                  MAIN0190
      STOP                                                                  MAIN0210
      END                                                                   MAIN0220
      BLOCK DATA
      COMMON/ALAGD/G(50)
      COMMON/ALAGF/GC(25,50)
      COMMON/ALAGI/G2P(325)                                                  ALMAIN
      DATA G2P/325*0.0E0/
      DATA GC/1250*0.0E0/
      DATA G/50*0.0E0/
      END
      SUBROUTINE ALAGA(N,M,K,X,EPS,AKMIN,DFN,MAXFN,IPR1,IPR2,IW,MODE)       ALGA0010
      REAL X(1),EPS(1)                                                      ALGA0020
      COMMON/ALAGC/F,MM,KL,IS,MK,NU                                         ALGA0030
      COMMON/ALAGD/G(50)                                                    ALGA0040
      COMMON/ALAGE/C(150)                                                   ALGA0050
```

ORIGINAL PAGE IS
OF POOR QUALITY

MAIN PROGRAM

ALAG BUILT-IN SUBROUTINES

```
      COMMON/ALAGF/GC(1250)                                              ALGA0060
      COMMON/ALAGG/T(150)                                                ALGA0070
      COMMON/ALAGH/GP(50)                                                ALGA0080
      COMMON/ALAGI/G2P(325)                                              ALGA0090
      COMMON/ALAGJ/V(50)                                                 ALGA0100
      COMMON/ALAGK/WW(150)                                               ALGA0110
      COMMON/ALAGL/W(2500)                                               ALGA0120
      COMMON/ALAGM/ZZ(100)                                               ALGA0130
      COMMON/ALAGN/LT(100)                                               ALGA0140
      EXTERNAL ALAGZ                                                     ALGA0150
 1000 FORMAT(30I4)                                                       ALGA0160
 1001 FORMAT(8E15.7)                                                     ALGA0170
      NU=MAX0(25,N)                                                      ALGA0180
      IF(M.GT.50)NU=N                                                    ALGA0190
      NX=N                                                               ALGA0200
      ICS=M                                                              ALGA0210
      ICG=M+M                                                            ALGA0220
      IS=M                                                               ALGA0230
      IL=IS+M                                                            ALGA0240
      IP=M                                                               ALGA0250
      ILT=M                                                              ALGA0260
      NM=M*(N+1)/2                                                       ALGA0270
      MM=M                                                               ALGA0280
      KL=K                                                               ALGA0290
      MINS=0                                                             ALGA0300
      AK=1E60                                                            ALGA0310
      R=1.                                                               ALGA0320
      MK=0                                                               ALGA0330
      DO 1 I=1,M                                                         ALGA0340
    1 C(ICS+I)=1.                                                        ALGA0350
      CALL ALAGB(N,M,X)                                                  ALGA0360
        WRITE(6,10000)(C(I),I=1,17)
10000   FORMAT(4X,5(E14.7,5X))
      DF=DFN                                                             ALGA0370
      IF(DFN.LT.0E0)DF=ABS(DFN*F)                                        ALGA0380
      IF(ABS(DFN).LT.1.0E-30) DF=F                                       ALGA0390
      IF(DF.LE.0.)DF=1.                                                  ALGA0400
      DO 2 I=1,M                                                         ALGA0410
      CC=C(I)                                                            ALGA0420
      IF(I.GT.K)CC=AMIN1(CC,0E0)                                         ALGA0430
      IF(ABS(CC).GT.C(ICS+I))C(ICS+I)=ABS(CC)                            ALGA0440
    2 CONTINUE                                                           ALGA0450
      IF(IPR1.EQ.0)GOTO4                                                 ALGA0460
      IF(MOD(MINS,IPR1).NE.0)GOTO4                                       ALGA0470
      PRINT 1002                                                         ALGA0480
 1002 FORMAT('1ENTRY TO ALAGA'///'0CONSTRAINT SCALE PARAMETERS ARE')     ALGA0490
      PRINT 1001,(C(ICS+I),I=1,M)                                        ALGA0500
    4 CONTINUE                                                           ALGA0510
      IF(MODE.LT.0)GOTO5                                                 ALGA0520
      DO 3 I=1,M                                                         ALGA0530
      T(IS+I)=2E0*DF/C(ICS+I)**2                                         ALGA0540
    3 T(I)=0.                                                            ALGA0550
    5 CONTINUE                                                           ALGA0560
      MO=IABS(MODE)                                                      ALGA0570
    8 CONTINUE                                                           ALGA0580
```


ORIGINAL PAGE IS
OF POOR QUALITY

```
      MINS=MINS+1                                                       ALGA0590
      DO 6 I=1,NN                                                       ALGA0600
    6 W(I)=G2P(I)                                                       ALGA0610
      IF(IPR1.EQ.0)GOTO7                                                ALGA0620
      IF(MOD(MINS,IPR1).NE.0)GOTO7                                      ALGA0630
      PRINT 1003,MINS                                                   ALGA0640
 1003 FORMAT(////'0OUTER ITERATION NUMBER IS',I3)                       ALGA0650
      PRINT 1004                                                        ALGA0660
 1004 FORMAT('0X(I)')                                                   ALGA0670
      PRINT 1001,(X(I),I=1,N)                                           ALGA0680
      PRINT 1005                                                        ALGA0690
 1005 FORMAT('0THETA(I)')                                               ALGA0700
      PRINT 1001,(T(I),I=1,M)                                           ALGA0710
      PRINT 1006                                                        ALGA0720
 1006 FORMAT('0SIGMA(I)')                                               ALGA0730
      PRINT 1001,(T(IS+I),I=1,M)                                        ALGA0740
    7 CONTINUE                                                          ALGA0750
      CALL CNMTA(ALAGI,N,X,PHI,GP,A,WW,DF,EPS,MD,MAXFN,IPR2,IEXIT)      ALGA0760
      CALL ALAGB(N,M,X)                                                 ALGA0770
      MD=3                                                              ALGA0780
      AKK=0.                                                            ALGA0790
      DO 10 I=1,M                                                       ALGA0800
      CC=C(I)                                                           ALGA0810
      IF(I.GT.K.AND.C(I).GE.T(I))CC=AMIN1(CC,0E0)                       ALGA0820
      T(I)=T(I)*T(IS+I)                                                 ALGA0830
      WW(I)=ABS(CC)/C(ICS+I)                                            ALGA0840
      IF(WW(I).GT.AKK)AKK=WW(I)                                         ALGA0850
   10 CONTINUE                                                          ALGA0860
      IF(IPR1.EQ.0)GOTO16                                               ALGA0870
      IF(MOD(MINS,IPR1).NE.0)GOTO16                                     ALGA0880
      PRINT 1007                                                        ALGA0890
 1007 FORMAT('0EXIT FROM CNMTA'/'0LAGRANGE MULTIPLIER ESTIMATES')       ALGA0900
      PRINT 1001,(T(I),I=1,M)                                           ALGA0910
      PRINT 1008                                                        ALGA0920
 1008 FORMAT('0LARGEST SCALED CONSTRAINT VIOLATION'/                    ALGA0930
     1' THIS ITERATION, BEST ITERATION')                                ALGA0940
      PRINT 1001,AKK,AK                                                 ALGA0950
      PRINT 1009                                                        ALGA0960
 1009 FORMAT('0CONSTRAINT RESIDUALS')                                   ALGA0970
      PRINT 1001,(C(I),I=1,M)                                           ALGA0980
      PRINT 1010                                                        ALGA0990
 1010 FORMAT('0SCALED CONSTRAINT VIOLATIONS')                           ALGA1000
      PRINT 1001,(WW(I),I=1,M)                                          ALGA1010
   16 CONTINUE                                                          ALGA1020
      IF(IEXIT.EQ.0  .OR.  IEXIT.EQ.3) GO TO 20                         ALGA1030
      IF(AKK.LE.AKMIN)GOTO20                                            ALGA1040
      IF(AKK.GE.AK)GOTO11                                               ALGA1050
      DO 19 I=1,NN                                                      ALGA1060
   19 G2P(I)=W(I)                                                       ALGA1070
      DO 17 I=1,M                                                       ALGA1080
      IF(I.GT.K.AND. ABS(T(I)     ).LT.1.0E-30.AND.C(I).GE.0E0) GO TO 17
      ZZ(IP+I)=-T(IS+I)*C(I)                                            ALGA1100
      IF(I.GT.K.AND.ZZ(IP+I).LT.-T(I))ZZ(IP+I)=-T(I)                    ALGA1120
   17 CONTINUE                                                          ALGA1130
      IF(MINS.EQ.1)GOTO40                                               ALGA1140
```


ORIGINAL PAGE IS
OF POOR QUALITY

```
      GOTO18                                                                    ALGA1150
   11 CONTINUE                                                                  ALGA1160
      IF(ABS(AKK-AK).LT.1.0E-10) GO TO 101
      GO TO 105
  101 DO 102 I = 1,M
      IF(C(ICB+I).GE.4.0*WW(I)) GO TO 102
      DS=9E0*T(IS+I)
      T(IS+I)=1E1*T(IS+I)
      IF(IPR1.NE.0)PRINT 1011,I,T(IS+I)
      DO 103J=1,N
  103 V(J)=GC((I-1)*NU+J)
      CALL MULDA(G2P,N,V,DS,V,N,N,DS)
  102 CONTINUE
      GO TO 18
  105 CONTINUE
      DO 14 I=1,M                                                               ALGA1170
      IF(WW(I).LE.AK.OR.C(ICB+I).GE.4E0*WW(I))GOTO14                            ALGA1180
      DS=9E0*T(IS+I)                                                            ALGA1190
      T(IS+I)=1E1*T(IS+I)                                                       ALGA1200
      IF(IPR1.NE.0)PRINT 1011,I,T(IS+I)                                         ALGA1210
 1011 FORMAT('0SIGMA(',I3,')  INCREASED TO ',E15.7)                             ALGA1220
      DO 12 J=1,N                                                               ALGA1230
   12 V(J)=GC((I-1)*NU+J)                                                       ALGA1240
      CALL MULDA(G2P,N,V,DS,V,N,N,DS)                                           ALGA1250
   14 CONTINUE                                                                  ALGA1260
   18 CONTINUE                                                                  ALGA1270
      DO 13 I=1,N                                                               ALGA1280
      IF(ABS(X(I)-G(IX+I)).GT.EPS(I))GOTO21                                     ALGA1290
   13 CONTINUE                                                                  ALGA1300
      PRINT 1013                                                                ALGA1310
 1013 FORMAT('0REQUESTED ACCURACY NOT OBTAINED')                                ALGA1320
   20 CONTINUE                                                                  ALGA1330
      IF(IEXIT.EQ.0)PRINT 2000                                                  ALGA1340
 2000 FORMAT('0MATRIX SET IN G2P BY USER IS NOT POSITIVE DEFINITE')             ALGA1350
      IF(IPR1.EQ.0)RETURN                                                       ALGA1360
      PRINT 1012                                                                ALGA1370
 1012 FORMAT('0BEST SOLUTION OBTAINED'/'0F,(G(I),I=1,N)')                       ALGA1380
      PRINT 1001,F,(G(I),I=1,N)                                                 ALGA1390
      RETURN                                                                    ALGA1400
   21 CONTINUE                                                                  ALGA1410
      IF(AKK.LT.AK)GOTO40                                                       ALGA1420
      DO 32 I=1,M                                                               ALGA1430
   32 V(I)=T(IL+I)                                                              ALGA1440
      GOTO70                                                                    ALGA1450
   40 CONTINUE                                                                  ALGA1460
      MK=0                                                                      ALGA1470
      KK=0                                                                      ALGA1480
      DO 41 I=1,M                                                               ALGA1490
      T(IL+I)=T(I)                                                              ALGA1500
      C(ICB+I)=WW(I)                                                            ALGA1510
      IF(I.GT.K.AND. ABS(T(IL+I)  ).LT.1.0E-30.AND.C(I).GE.0E0) GO TO 41
      KK=KK+1                                                                   ALGA1530
      LT(ILT+KK)=I                                                              ALGA1540
      GP(KK)=-1E30                                                              ALGA1550
      IF(I.GT.K)GP(KK)=-T(IL+I)                                                 ALGA1560
```


ORIGINAL PAGE IS
OF POOR QUALITY

```
      V(KK)=1E30                                                        ALGA1570
      ZZ(KK)=-C(I)                                                      ALGA1580
   41 CONTINUE                                                          ALGA1590
      IF(KK.EQ.0)GOTO20                                                 ALGA1600
      DO 42 I=1,N                                                       ALGA1610
   42 G(IX+I)=X(I)                                                      ALGA1620
      KKK=KK*(KK+1)/2                                                   ALGA1630
      II=MAXO(KKK+NN,KK*KK)                                             ALGA1640
      IF(II.LE.IW)GOTO50                                                ALGA1650
      PRINT 2001,II                                                     ALGA1660
2001  FORMAT('0INCREASE STORAGE IN COMMON/ALAGL TO',I7,'ELEMENTS')      ALGA1670
      RETURN                                                            ALGA1680
   50 CONTINUE                                                          ALGA1690
      II=IW-KKK                                                         ALGA1700
      DO 53 I=1,KK                                                      ALGA1710
      LI=LT(ILT+I)                                                      ALGA1720
      DO 51 JJ=1,N                                                      ALGA1730
   51 X(JJ)=GC((LI-1)*NU+JJ)                                            ALGA1740
      CALL MULDE(W,N,X,X,N)                                             ALGA1750
      DO 53 J=1,I                                                       ALGA1760
      LJ=LT(ILT+J)                                                      ALGA1770
      Z=0.                                                              ALGA1780
      DO 52 JJ=1,N                                                      ALGA1790
   52 Z=Z+X(JJ)*GC((LJ-1)*NU+JJ)                                        ALGA1800
      II=II+1                                                           ALGA1810
   53 W(II)=Z                                                           ALGA1820
      JJ=IW-KKK                                                         ALGA1830
      II=0                                                              ALGA1840
      DO 56 I=1,KK                                                      ALGA1850
      DO 55 J=1,I                                                       ALGA1860
      JJ=JJ+1                                                           ALGA1870
   55 W(II+J)=W(JJ)                                                     ALGA1880
   56 II=II+KK                                                          ALGA1890
      CALL BCDMA(KK,W,KK,ZZ,GP,V,T,Z,LT,JJ,WW)                          ALGA1900
      IF(IPR1.EQ.0)GOTO59                                               ALGA1910
      IF(MOD(MINS,IPP1).NE.0)GOTO59                                     ALGA1920
      PRINT 1020,KK                                                     ALGA1930
1020  FORMAT(I4,' ACTIVE CONSTRAINTS, NUMBERED')                        ALGA1940
      PRINT 1000,(LT(ILT+I),I=1,KK)                                     ALGA1950
      PRINT 1021                                                        ALGA1960
1021  FORMAT('0LAGRANGE MULTIPLIER CORRECTIONS FOR ACTIVE CONSTRAINTS') ALGA1970
      PRINT 1001,(T(I),I=1,KK)                                          ALGA1980
   59 CONTINUE                                                          ALGA1990
      DO 60 I=1,M                                                       ALGA2000
   60 V(I)=T(IL+I)                                                      ALGA2010
      DO 62 I=1,KK                                                      ALGA2020
      LI=LT(ILT+I)                                                      ALGA2030
      V(LI)=V(LI)+T(I)                                                  ALGA2040
      IF(ABS(ZZ(IP+LI)).LE.1.0E-30) GO TO 62                            ALGA2050
      Z=4E0*ABS((T(I)-ZZ(IP+LI))/ZZ(IP+LI))                             ALGA2060
      IF(Z.LE.1E0)GOTO62                                                ALGA2070
      DS=(Z-1E0)*T(IS+LI)                                               ALGA2080
      T(IS+LI)=Z*T(IS+LI)                                               ALGA2090
      IF(IPR1.NE.0)PRINT 1011,LI,T(IS+LI)                               ALGA2100
      DO 61 J=1,N
```


ORIGINAL PAGE IS
OF POOR QUALITY

```
   61 GP(J)=GG((LI-1)*NU+J)                                                ALGA2110
      CALL ADDA(G2P,N,GP,TS,GP,M,N,CC)                                     ALGA2120
   62 CONTINUE                                                             ALGA2130
      AK=AKK                                                               ALGA2140
   70 CONTINUE                                                             ALGA2150
      DO 71 I=1,M                                                          ALGA2160
   71 T(I)=V(I)/T(IS+I)                                                    ALGA2170
      DO 72 I=1,N                                                          ALGA2180
   72 X(I)=G(IX+I)                                                         ALGA2190
      CF=1E50                                                              ALGA2200
      GOTO 8                                                               ALGA2210
      END                                                                  ALGA2220
      SUBROUTINE ALAGZ(N,X,PHI,GPHI)                                       ALGZ0010
      REAL X(1),GPHI(1)                                                    ALGZ0020
      COMMON/ALAGC/F,M,K,IS,MK,NU                                          ALGZ0030
      COMMON/ALAGD/G(50)                                                   ALGZ0040
      COMMON/ALAGE/C(150)                                                  ALGZ0050
      COMMON/ALAGF/GC(1250)                                                ALGZ0060
      COMMON/ALAGG/T(150)                                                  ALGZ0070
      IF(MK.EQ.1)CALL ALAGB (N,M,X)                                        ALGZ0080
      MK=1                                                                 ALGZ0090
      PHI=0.                                                               ALGZ0100
      DO 10 I=1,N                                                          ALGZ0110
   10 GPHI(I)=G(I)                                                         ALGZ0120
      DO 12 I=1,M                                                          ALGZ0130
      CC=C(I)-T(I)                                                         ALGZ0140
      IF(I.GT.K)CC=AMIN1(CC,0E0)                                           ALGZ0150
      Y=T(IS+I)*CC                                                          ALGZ016
      IF( ABS(Y).LT.1.0E-30) GO TO 12                                       ALGZ018
      PHI=PHI+Y*CC                                                         ALGZ0190
      DO 11 J=1,N                                                          ALGZ0200
   11 GPHI(J)=GPHI(J)+Y*GC((I-1)*NU+J)                                     ALGZ0210
   12 CONTINUE                                                             ALGZ0220
      PHI=.5E0*PHI+F                                                       ALGA0230
      RETURN                                                               ALGZ0240
      END                                                                  CNTA0010
      SUBROUTINE QNWTA(FUNCT,N,X,F,G,H,W,CFN,EPS,MODE,MAXFN,IPRINT,        CNTA0020
     1IEXIT)                                                               CNTA0030
      REAL X(1),G(1),H(1),W(1),EPS(1)                                      CNTA0040
      CALL FUNCT(N,X,F,G)                                                  CNTA0050
      IF(IPRINT.NE.0)PRINT 1000                                            CNTA0060
 1000 FORMAT('1ENTRY TO QNWTA'/)                                           CNTA0070
      NN=N*(N+1)/2                                                         CNTA0080
      IG=N                                                                 CNTA0090
      IGG=N+N                                                              CNTA0100
      IS=IGG                                                               CNTA0110
      IEXIT=0                                                              CNTA0120
      IR=N                                                                 CNTA0130
      IF(MODE.EQ.3)GOTO15                                                  CNTA0140
      IF(MODE.EQ.2)GOTO10                                                  CNTA0150
      IJ=NN+1                                                              CNTA0160
      DO 5 I=1,N                                                           CNTA0170
      DO 6 J=1,I                                                           CNTA0180
      IJ=IJ-1
    6 H(IJ)=0.
```


ORIGINAL PAGE IS
OF POOR QUALITY

ORIGINAL PAGE IS
OF POOR QUALITY

```
    5 H(IJ)=1.                                                          CNTA0190
      GOTO15                                                            CNTA0200
   10 CONTINUE                                                          CNTA0210
      CALL MULDB(H,N,IR)                                                CNTA0220
      IF(IR.LT.N)RETURN                                                 CNTA0230
   15 CONTINUE                                                          CNTA0240
      Z=F                                                               CNTA0250
      ITN=0                                                             CNTA0260
      CALL FUNCT(N,X,F,G)                                               CNTA0270
      IFN=1                                                             CNTA0280
      DF=DFN                                                            CNTA0290
      IF(ABS(DFN).LT.1.0E-30) DF=F-Z                                    CNTA0300
      IF(DFN.LT.0.)DF=ABS(DF*F)                                         CNTA0310
      IF(DF.LE.0.)DF=1.                                                 CNTA0320
   20 CONTINUE                                                          CNTA0330
      IF(IPRINT.EQ.0)GOTO21                                             CNTA0340
      IF(MOD(ITN,IPRINT).NE.0)GOTO21                                    CNTA0350
      PRINT 1001,ITN,IFN                                                CNTA0360
 1001 FORMAT(24I5)                                                      CNTA0370
      PRINT 1002,F                                                      CNTA0380
 1002 FORMAT((8E15.7))                                                  CNTA0390
      IF(IPRINT.LT.0)GOTO21                                             CNTA0400
      PRINT 1002,(X(I),I=1,N)                                           CNTA0410
      PRINT 1002,(G(I),I=1,N)                                           CNTA0420
   21 CONTINUE                                                          CNTA0430
      ITN=ITN+1                                                         CNTA0440
      DO 22 I=1,N                                                       CNTA0450
   22 W(IG+I)=G(I)                                                      CNTA0460
      CALL MULDE(F,N,G,W,IR)                                            CNTA0470
      GS=0.                                                             CNTA0480
      DO 29 I=1,N                                                       CNTA0490
      W(IS+I)=-G(I)                                                     CNTA0500
   29 GS=GS-G(I)*W(IG+I)                                                CNTA0510
C     IEXIT=2                                                           CNTA0520
C     IF(GS.GE.0.)GOTO92                                                CNTA0530
      GSO=GS                                                            CNTA0540
      ALPHA=-2.*DF/GS                                                   CNTA0550
      IF(ALPHA.GT.1.)ALPHA=1.                                           CNTA0560
      DF=F                                                              CNTA0570
      TOT=0.                                                            CNTA0580
   30 CONTINUE                                                          CNTA0590
      IEXIT=3                                                           CNTA0600
      IF(IFN.EQ.MAXFN)GOTO92                                            CNTA0610
      IEXIT=4
      IF(IFN.GT.MAXFN) GO TO 92
      ICON=0                                                            CNTA0620
      IEXIT=1                                                           CNTA0630
      DO 31 I=1,N                                                       CNTA0640
      Z=ALPHA*W(IS+I)                                                   CNTA0650
      IF(ABS(Z).GE.EPS(I))ICON=1                                        CNTA0660
   31 X(I)=X(I)+Z                                                       CNTA0670
      CALL FUNCT(N,X,FY,G)                                              CNTA0680
      IFN=IFN+1                                                         CNTA0690
      GYS=0.                                                            CNTA0700
      DO 32 I=1,N                                                       CNTA0710
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
   32 GYS=GYS+G(I)*W(IS+I)                                          CNTA0720
      IF(FY.GE.F)GOTO40                                             CNTA0730
      IF(ABS(GYS/GS0).LE..9)GOTO50                                  CNTA0740
      IF(GYS.GT.0.)GOTO40                                           CNTA0750
      TOT=TOT+ALPHA                                                 CNTA0760
      Z=10.                                                         CNTA0770
      IF(GS.LT.GYS)Z=GYS/(GS-GYS)                                   CNTA0780
      IF(Z.GT.10.)Z=10.                                             CNTA0790
      ALPHA=ALPHA*Z                                                 CNTA0800
      F=FY                                                          CNTA0810
      GS=GYS                                                        CNTA0820
      GOTO30                                                        CNTA0830
   40 CONTINUE                                                      CNTA0840
      DO 41 I=1,N                                                   CNTA0850
   41 X(I)=X(I)-ALPHA*W(IS+I)                                       CNTA0860
      IF(ICON.EQ.0)GOTO92                                           CNTA0870
      Z=3.*(F-FY)/ALPHA+GYS+GS                                      CNTA0880
      IF(ABS(Z).GT.1.0E35) GO TO 36
      GO TO 37
36    Z=1.0
      GO TO 39
37    CONTINUE
      ZZ=SQRT(Z**2-GS*GYS)                                          CNTA0890
      Z=1.-(GYS+ZZ-Z)/(2.*ZZ+GYS-GS)                                CNTA0900
39    CONTINUE
      ALPHA=ALPHA*Z                                                 CNTA0910
      GOTO20                                                        CNTA0920
   50 CONTINUE                                                      CNTA0930
      ALPHA=TOT+ALPHA                                               CNTA0940
      F=FY                                                          CNTA0950
      IF(ICON.EQ.0)GOTO90                                           CNTA0960
      DF=DF-F                                                       CNTA0970
      DGS=GYS-GS0                                                   CNTA0980
      DO 51 I=1,N                                                   CNTA0990
      W(IGG+I)=G(I)                                                 CNTA1000
   51 G(I)=-W(IG+I)                                                 CNTA1010
      IF(DGS+ALPHA*GS0.GT.0.)GOTO60                                 CNTA1020
C     COMPLEMENTARY DFP FORMULA                                       QNTA10
      SIG=1./GS0                                                    CNTA1040
      IR=-IR                                                        CNTA1050
      CALL MULDA(H,N,G,SIG,W,IR,1,0.)                               CNTA1060
      DO 52 I=1,N                                                   CNTA1070
   52 G(I)=W(IGG+I)-W(IG+I)                                         CNTA1080
      SIG=1./(ALPHA*DGS)                                            CNTA1090
      IR=-IR                                                        CNTA1100
      CALL MULDA(H,N,G,SIG,W,IR,0,0.)                               CNTA1110
      GOTO70                                                        CNTA1120
   60 CONTINUE                                                      CNTA1130
C     DFP FORMULA                                                     QNTA11
      ZZ=ALPHA/(DGS-ALPHA*GS0)                                      CNTA1150
      SIG=-ZZ                                                       CNTA1160
      CALL MULDA(H,N,G,SIG,W,IR,1,1E-7)                             CNTA1170
      Z=DGS*ZZ-1.                                                   CNTA1180
      DO 61 I=1,N                                                   CNTA1190
   61 G(I)=W(IGG+I)+Z*W(IG+I)                                       CNTA1200
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      SIG=1./(ZZ*DGS**2)                                                CNTA1210
      CALL MULDA(H,N,G,SIG,W,IR,0,0.)                                   CNTA1220
   70 CONTINUE                                                          CNTA1230
      DO 71 I=1,N                                                       CNTA1240
   71 G(I)=W(IGG+I)                                                     CNTA1250
      GOTO20                                                            CNTA1260
   92 CONTINUE                                                          CNTA1270
      DO 91 I=1,N                                                       CNTA1280
   91 G(I)=W(IG+I)                                                      CNTA1290
   90 CONTINUE                                                          CNTA1300
      IF(IPRINT.EQ.0)RETURN                                             CNTA1310
      PRINT 1001,ITN,IFN,IEXIT                                          CNTA1320
      PRINT 1002,F                                                      CNTA1330
      PRINT 1002,(X(I),I=1,N)                                           CNTA1340
      PRINT 1002,(G(I),I=1,N)                                           CNTA1350
      RETURN                                                            CNTA1360
      END                                                               CNTA1370
      SUBROUTINE MULDA(A,N,Z,SIG,W,IR,MK,EPS)                           MULDA010
      DIMENSION A(1),Z(1),W(1)                                          MULDE0020
C     UPDATE FACTORS GIVEN IN A BY    SIG*Z*ZTRANSPOSE                    MULDA000
      IF(N.GT.1)GOTO1                                                   MULDA0040
      A(1)=A(1)+SIG*Z(1)**2                                             MULDA0050
      IR=1                                                              MULDA0060
      IF(A(1).GT.0.)RETURN                                              MULDA0070
      A(1)=0.                                                           MULDA0080
      IR=0                                                              MULDA0090
      RETURN                                                            MULDA0100
    1 CONTINUE                                                          MULDA0110
      NP=N+1                                                            MULDA0120
      IF(SIG.GT.0.)GOTO40                                               MULDA0130
      IF( ABS(SIG).LT.1.0E-30 .OR.IR.EQ.0) RETURN                       MULDA0140
      TI=1./SIG                                                         MULDA0150
      IJ=1                                                              MULDA0160
      IF(MK.EQ.0)GOTO10                                                 MULDA0170
      DO 7 I=1,N                                                        MULDA0180
      IF( ABS(A(IJ)).GT.1.0E-30) TI=TI+W(I)**2/A(IJ)                    MULDA0190
    7 IJ=IJ+NP-I                                                        MULDA0200
      GOTO20                                                            MULDA0210
   10 CONTINUE                                                          MULDA0220
      DO 11 I=1,N                                                       MULDA0230
   11 W(I)=Z(I)                                                         MULDA0240
      DO 15 I=1,N                                                       MULDA0250
      IP=I+1                                                            MULDA0260
      V=W(I)                                                            MULDA0270
      IF(A(IJ).GT.0.)GOTO12                                             MULDA0280
      W(I)=0.                                                           MULDA0290
      IJ=IJ+NP-I                                                        MULDA0300
      GOTO15                                                            MULDA0310
   12 CONTINUE                                                          MULDA0320
      TI=TI+V**2/A(IJ)                                                  MULDA0330
      IF(I.EQ.N)GOTO14                                                  MULDA0340
      DO 13 J=IP,N                                                      MULDA0350
      IJ=IJ+1                                                           MULDA0360
   13 W(J)=W(J)-V*A(IJ)                                                 MULDA0370
   14 IJ=IJ+1                                                           MULDA0380
```

```
15  CONTINUE                                                          MUDA0390
20  CONTINUE                                                          MUDA0400
    IF(TI.LE.0.)GOTO21                                                MUDA0410
    IF(TI.GT.0.)GOTO22                                                MUDA0420
    IF(MK-1)40,40,23                                                  MUDA0430
21  TI=0.                                                             MUDA0440
    IR=-IR-1                                                          MUDA0450
    GOTO23                                                            MUDA0460
22  TI=EPS/SIG                                                        MUDA0470
    IF( ABS(EPS).LT.1.0E-30) IR=IR-1                                  MUDA0480
23  CONTINUE                                                          MUDA0490
    MM=1                                                              MUDA0500
    TIM=TI                                                            MUDA0510
    DO 30 I=1,N                                                       MUDA0520
    J=NP-I                                                            MUDA0530
    IJ=IJ-I                                                           MUDA0540
    IF( ABS(A(IJ)).GT.1.0E-30)TIM=TI-W(J)**2/A(IJ)                    MUDA0550
    W(J)=TI                                                           MUDA0560
30  TI=TIM                                                            MUDA0570
    GOTO41                                                            MUDA0580
40  CONTINUE                                                          MUDA0590
    MM=0                                                              MUDA0600
    TIM=1./SIG                                                        MUDA0610
41  CONTINUE                                                          MUDA0620
    IJ=1                                                              MUDA0630
    DO 66 I=1,N                                                       MUDA0640
    IP=I+1                                                            MUDA0650
    V=Z(I)                                                            MUDA0660
    IF(A(IJ).GT.0.)GOTO53                                             MUDA0670
    IF(IR.GT.0 .OR.SIG.LT.0..OR. ABS(V).LT.1.0E-30) GO TO 52          MUDA0680
    IR=1-IR                                                           MUDA0690
    A(IJ)=V**2/TIM                                                    MUDA0700
    IF(I.EQ.N)RETURN                                                  MUDA0710
    DO 51 J=IP,N                                                      MUDA0720
    IJ=IJ+1                                                           MUDA0730
51  A(IJ)=Z(J)/V                                                      MUDA0740
    RETURN                                                            MUDA0750
52  CONTINUE                                                          MUDA0760
    TI=TIM                                                            MUDA0770
    IJ=IJ+NP-I                                                        MUDA0780
    GOTO66                                                            MUDA0790
53  CONTINUE                                                          MUDA0800
    AL=V/A(IJ)                                                        MUDA0810
    IF(MM)54,54,55                                                    MUDA0820
54  TI=TIM+V*AL                                                       MUDA0830
    GOTO56                                                            MUDA0840
55  TI=W(I)                                                           MUDA0850
56  CONTINUE                                                          MUDA0860
    R=TI/TIM                                                          MUDA0870
    A(IJ)=A(IJ)*R                                                     MUDA0880
    IF( ABS(R).LT.1.0E-30) GO TO 70                                   MUDA0890
    IF(I.EQ.N)GOTO70                                                  MUDA0900
    B=AL/TI                                                           MUDA0910
    IF(R.GT.4.)GOTO62                                                 MUDA0920
    DO 61 J=IP,N                                                      MUDA0930
```


ORIGINAL PAGE IS
OF POOR QUALITY

ORIGINAL PAGE IS
OF POOR QUALITY

```
      IJ=IJ+1                                                          MUDA0940
      Z(J)=Z(J)-V*A(IJ)                                                MUDA0950
   61 A(IJ)=A(IJ)+B*Z(J)                                               MUDA0960
      GOTO64                                                           MUDA0970
   62 GM=TIM/TI                                                        MUDA0980
      DO 63 J=IP,N                                                     MUDA0990
      IJ=IJ+1                                                          MUDA1000
      Y=A(IJ)                                                          MUDA1010
      A(IJ)=R*Z(J)+Y*GM                                                MUDA1020
   63 Z(J)=Z(J)-V*Y                                                    MUDA1030
   64 CONTINUE                                                         MUDA1040
      TIM=TI                                                           MUDA1050
      IJ=IJ+1                                                          MUDA1060
   66 CONTINUE                                                         MUDA1070
   70 CONTINUE                                                         MUDA1080
      IF(IR.LT.0)IR=-IR                                                MUDA1090
      RETURN                                                           MUDA1100
      END                                                              MUDA1110
      SUBROUTINE MULDB(A,N,IR)                                         MUDB0010
      DIMENSION A(1)                                                   MUDB0020
      IR=N                                                             MUDB0030
      IF(N.GT.1)GOTO100                                                MUDB0040
      IF(A(1).GT.0.)RETURN                                             MUDB0050
      A(1)=0.                                                          MUDB0060
      IR=0                                                             MUDB0070
      RETURN                                                           MUDB0080
  100 CONTINUE                                                         MUDB0090
      NP=N+1                                                           MUDB0100
      II=1                                                             MUDB0110
      DO 104 I=2,N                                                     MUDB0120
      AA=A(II)                                                         MUDB0130
      NI=II+NP-I                                                       MUDB0140
      IF(AA.GT.0.)GOTO101                                              MUDB0150
      A(II)=0.                                                         MUDB0160
      IR=IR-1                                                          MUDB0170
      II=NI+1                                                          MUDB0180
      GOTO104                                                          MUDB0190
  101 CONTINUE                                                         MUDB0200
      IP=II+1                                                          MUDB0210
      II=NI+1                                                          MUDB0220
      JK=II                                                            MUDB0230
      DO 103 IJ=IP,NI                                                  MUDB0240
      V=A(IJ)/AA                                                       MUDB0250
      DO 102 IK=IJ,NI                                                  MUDB0260
      A(JK)=A(JK)-A(IK)*V                                              MUDB0270
  102 JK=JK+1                                                          MUDB0280
  103 A(IJ)=V                                                          MUDB0290
  104 CONTINUE                                                         MUDB0300
      IF(A(II).GT.0.)RETURN                                            MUDB0310
      A(II)=0.                                                         MUDB0320
      IR=IR-1                                                          MUDB0330
      RETURN                                                           MUDB0340
      END                                                              MUDB0350
      SUBROUTINE MULDE(A,N,Z,W,IR)                                     MUDE0010
      DIMENSION A(1),Z(1),W(1)                                         MUDA0020
```

```
C       MULTIPLY A VECTOR Z BY THE INVERSE OF THE FACTORS GIVEN IN A      MUDE00
        IF(IP.LT.N)RETURN                                                 MUDE0040
        W(1)=Z(1)                                                         MUDE0050
        IF(N.GT.1)GOTO400                                                 MUDE0060
        Z(1)=Z(1)/A(1)                                                    MUDE0070
        RETURN                                                            MUDE0080
  400   CONTINUE                                                          MUDE0090
        DO 402 I=2,N                                                      MUDE0100
        IJ=I                                                              MUDE0110
        I1=I-1                                                            MUDE0120
        V=Z(I)                                                            MUDE0130
        DO 401 J=1,I1                                                     MUDE0140
        V=V-A(IJ)*Z(J)                                                    MUDE0150
  401   IJ=IJ+N-J                                                         MUDE0160
        W(I)=V                                                            MUDE0170
  402   Z(I)=V                                                            MUDE0180
        Z(N)=Z(N)/A(IJ)                                                   MUDE0190
        NP=N+1                                                            MUDE0200
        DO 411 NIP=2,N                                                    MUDE0210
        I=NP-NIP                                                          MUDE0220
        II=IJ-NIP                                                         MUDE0230
        V=Z(I)/A(II)                                                      MUDE0240
        IP=I+1                                                            MUDE0250
        IJ=II                                                             MUDE0260
        DO 410 J=IP,N                                                     MUDE0270
        II=II+1                                                           MUDE0280
  410   V=V-A(II)*Z(J)                                                    MUDE0290
  411   Z(I)=V                                                            MUDE0300
        RETURN                                                            MUDE0310
        END                                                               MUDE0320
        SUBROUTINE BQDMA(N,A,IA,B,BL,BU,X,G,LT,K,G)                       BQDMA0010
        DIMENSION A(IA,1),B(1),BL(1),BU(1),X(1),LT(1),G(1)                BQDMA0020
        IS=N                                                              BQDMA0030
        IAS=N                                                             BQDMA0040
        IV=N                                                              BQDMA0050
        ICAC=N+N                                                          BQDMA0060
        ID=ICAC                                                           BQDMA0070
        DO 9 I=1,N                                                        BQDMA0080
    9   G(I)=-B(I)                                                        BQDMA0090
        DO 10 I=1,N                                                       BQDMA0100
        X(I)=0.                                                           BQDMA0110
        LT(I)=I                                                           BQDMA0120
        G(ICAC+I)=A(I,I)                                                  BQDMA0130
        IF(0..GE.BL(I).AND.0..LE.BU(I))GOTO10                             BQDMA0140
        IF(0..LT.BL(I))X(I)=BL(I)                                         BQDMA0150
        IF(0..GT.BU(I))X(I)=BU(I)                                         BQDMA0160
        DO 12 J=1,I                                                       BQDMA0170
   12   G(J)=G(J)+A(J,I)*X(I)                                             BQDMA0180
        IF(I.EQ.N)GOTO10                                                  BQDMA0190
        II=I+1                                                            BQDMA0200
        DO 11 J=II,N                                                      BQDMA0210
   11   G(J)=G(J)+A(I,J)*X(I)                                             BQDMA0220
   10   CONTINUE                                                          BQDMA0230
        K=0                                                               BQDMA0240
        K1=1                                                              BQDMA0250
```


ORIGINAL PAGE IS
OF POOR QUALITY

```
20  CONTINUE                                                              BQMA0260
    IOUT=0                                                                BQMA0270
    DEL=0.                                                                BQMA0280
    DO 21 I=K1,N                                                          BQMA0290
    LI=LT(I)                                                              BQMA0300
    IF( ABS(X(LI)-BL(I)).LT.1.0E-30.AND.G(I).GE.0.E0) GO TO 21
    IF( ABS(X(LI)-BL(I)).LT.1.0E-30.AND.G(I).LE.0.E0) GO TO 21
    IF(G(I).LT.0.)GOTO22                                                  BQMA0330
    Z=X(LI)-BL(LI)                                                        BQMA0340
    J=1                                                                   BQMA0350
    GOTO23                                                                BQMA0360
22  CONTINUE                                                              BQMA0370
    Z=BU(LI)-X(LI)                                                        BQMA0380
    J=0                                                                   BQMA0390
23  CONTINUE                                                              BQMA0400
    IF(C(ICAC+I).LE.0.)GOTO24                                             BQMA0410
    BETA=ABS(G(I))/G(ICAC+I)                                              BQMA0420
    IF(BETA.GE.Z)GOTO24                                                   BQMA0430
    Z=BETA                                                                BQMA0440
    D=.5*Z*ABS(G(I))                                                      BQMA0450
    J=-1                                                                  BQMA0460
    GOTO26                                                                BQMA0470
24  CONTINUE                                                              BQMA0480
    D=Z*(ABS(G(I))-.5*Z*G(ICAC+I))                                        BQMA0490
26  CONTINUE                                                              BQMA0500
    IF(D.LT.DEL)GOTO21                                                    BQMA0510
    DEL=D                                                                 BQMA0520
    ALPHA=Z                                                               BQMA0530
    IOUT=I                                                                BQMA0540
    IIN=I                                                                 BQMA0550
    IF(J.LT.0)IIN=0                                                       BQMA0560
    LB=J                                                                  BQMA0570
21  CONTINUE                                                              BQMA0580
    IF(IOUT.NE.0)GOTO29                                                   BQMA0590
27  CONTINUE                                                              BQMA0600
    Q=0.                                                                  BQMA0610
    DO 28 I=1,N                                                           BQMA0620
    LI=LT(I)                                                              BQMA0630
28  Q=Q+X(LI)*(G(I)-B(LI))                                                BQMA0640
    Q=.5*Q                                                                BQMA0650
    RETURN                                                                BQMA0660
29  CONTINUE                                                              BQMA0670
    SIG=1.                                                                BQMA0680
    IF(G(IOUT).GT.0.)SIG=-1.                                              BQMA0690
    LIOUT=LT(IOUT)                                                        BQMA0700
    LIIN=LIOUT                                                            BQMA0710
25  CONTINUE                                                              BQMA0720
    SAS=G(ICAC+IOUT)                                                      BQMA0730
    IF(K.EQ.0)GOTO31                                                      BQMA0740
    DO 30 I = 1, K
30  G(IS+I)=G(ID+I)*A(IOUT,I)                                             BQMA0770
31  CONTINUE                                                              BQMA0780
    DO 37 I=K1,N                                                          BQMA0790
    LI=LT(I)                                                              BQMA0800
    IF(LI-LIOUT)32,37,33
```


ORIGINAL PAGE IS
OF POOR QUALITY

ORIGINAL PAGE IS
OF POOR QUALITY

```
32  Z=A(LI,LIOUT)                                                    QMA0810
    GOTO34                                                           QMA0820
33  Z=A(LIOUT,LI)                                                    QMA0830
34  CONTINUE                                                         QMA0840
    IF(K.EQ.0)GOTO36                                                 QMA0850
    DO 35 J=1,K                                                      QMA0860
35  Z=Z-A(I,J)*G(IS+J)                                               QMA0870
36  G(IS+I)=Z                                                        QMA0880
37  CONTINUE                                                         QMA0890
    G(IS+IOUT)=SAS                                                   QMA0900
    IF(K.EQ.0)GOTO42                                                 QMA0910
    G(IS+K)=-A(IOUT,K)                                               QMA0920
    IF(K.EQ.1)GOTO42                                                 QMA0930
    I=K                                                              QMA0940
    DO 41 II=2,K                                                     QMA0950
    I=I-1                                                            QMA0960
    Z=-A(IOUT,I)                                                     QMA0970
    I1=I+1                                                           QMA0980
    DO 40 J=I1,K                                                     QMA0990
40  Z=Z-G(IS+J)*A(J,I)                                               QMA1000
41  G(IS+I)=Z                                                        QMA1010
42  CONTINUE                                                         QMA1020
    IF( ABS(SIG-1.).LT.1.0E-30) GO TO 51
    DO 50 I=1,N                                                      QMA1040
50  G(IS+I)=-G(IS+I)                                                 QMA1050
51  CONTINUE                                                         QMA1060
    IF(K.EQ.0)GOTO62                                                 QMA1070
    DO 61 I=1,K                                                      QMA1080
    IF( ABS(G(IS+I)).LT.1.0E-30) GO TO 61
    LI=LT(I)                                                         QMA1100
    J=1                                                              QMA1110
    Z=BL(LI)-X(LI)                                                   QMA1120
    IF(G(IS+I).LT.0.)GOTO60                                          QMA1130
    J=0                                                              QMA1140
    Z=BU(LI)-X(LI)                                                   QMA1150
60  CONTINUE                                                         QMA1160
    Z=Z/G(IS+I)                                                      QMA1170
    IF(Z.GE.ALPHA)GOTO61                                             QMA1180
    ALPHA=Z                                                          QMA1190
    LB=J                                                             QMA1200
    IIN=I                                                            QMA1210
    LIIN=LI                                                          QMA1220
61  CONTINUE                                                         QMA1230
62  CONTINUE                                                         QMA1240
    X(LIOUT)=X(LIOUT)+SIG*ALPHA                                      QMA1250
    IF(K.EQ.0)GOTO71                                                 QMA1260
    DO 70 I=1,K                                                      QMA1270
    LI=LT(I)                                                         QMA1280
70  X(LI)=X(LI)+ALPHA*G(IS+I)                                        QMA1290
71  CONTINUE                                                         QMA1300
    DO 72 I=K1,N                                                     QMA1310
72  G(I)=G(I)+ALPHA*G(IAS+I)                                         QMA1320
    IF(IIN.EQ.0)GOTO90                                               QMA1330
    X(LIIN)=BL(LIIN)                                                 QMA1340
    IF(LB.EQ.0)X(LIIN)=BU(LIIN)                                      QMA1350
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      IF(IIN.EQ.ICUT)GOTO20                                          QMA1360
      K2=K-1                                                         QMA1370
      SG=G(ID+IIN)                                                   QMA1380
      I1=IIN+1                                                       QMA1390
      DO  80  I=I1,N                                                 QMA1400
   80 G(IV+I)=A(I,IIN)                                               QMA1410
      IF(IIN.EQ.K)GOTO86                                             QMA1420
      I2=IIN+2                                                       QMA1430
      S0=1./SG                                                       QMA1440
      DO  85  I=IIN,K2                                               QMA1450
      V=G(IV+I1)                                                     QMA1460
      VD=V/G(ID+I1)                                                  QMA1470
      S1=S0+V*VD                                                     QMA1480
      R=S1/S0                                                        QMA1490
      G(ID+I)=G(ID+I1)*R                                             QMA1500
      BETA=VD/S1                                                     QMA1510
      IF(R.GT.4.)GOTO841                                             QMA1520
      DO  81  J=I2,N                                                 QMA1530
   81 G(IV+J)=G(IV+J)-V*A(J,I1)                                      QMA1540
      IF(I1.GT.K2)GOTO83                                             QMA1550
      DO  82  J=I1,K2                                                QMA1560
      J1=J+1                                                         QMA1570
   82 A(J,I)=A(J1,I1)+BETA*G(IV+J1)                                  QMA1580
   83 CONTINUE                                                       QMA1590
      A(K,I)=BETA                                                    QMA1600
      DO  84  J=K1,N                                                 QMA1610
   84 A(J,I)=A(J,I1)+BETA*G(IV+J)                                    QMA1620
      GOTO849                                                        QMA1630
  841 CONTINUE                                                       QMA1640
      IF(I1.GT.K2)GOTO843                                            QMA1650
      DO  842  J=I1,K2                                               QMA1660
      J1=J+1                                                         QMA1670
  842 A(J,I)=BETA*G(IV+J1)+A(J1,I1)/R                                QMA1680
  843 CONTINUE                                                       QMA1690
      A(K,I)=BETA                                                    QMA1700
      DO  844  J=K1,N                                                QMA1710
  844 A(J,I)=BETA*G(IV+J)+A(J,I1)/R                                  QMA1720
      DO  845  J=I2,N                                                QMA1730
  845 G(IV+J)=G(IV+J)-V*A(J,I1)                                      QMA1740
  849 CONTINUE                                                       QMA1750
      LT(I)=LT(I1)                                                   QMA1760
      S0=S1                                                          QMA1770
      I1=I2                                                          QMA1780
   85 I2=I2+1                                                        QMA1790
      SG=1./S1                                                       QMA1800
      LT(K)=LIIN                                                     QMA1810
      G(ID+K)=SG                                                     QMA1820
      IF(IIN.EQ.1)GOTO851                                            QMA1830
      II=IIN-1                                                       QMA1840
      DO  852  I=1,II                                                QMA1850
      Z=A(IIN,I)                                                     QMA1860
      DO  853  J=IIN,K2                                              QMA1870
  853 A(J,I)=A(J+1,I)                                                QMA1880
  852 A(K,I)=Z                                                       QMA1890
  851 CONTINUE                                                       QMA1900
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
 86  CONTINUE                                                          QMA1910
     DO 87 I=K1,N                                                      QMA1920
 87  G(ICAC+I)=G(ICAC+I)+SG*G(IV+I)**2                                 QMA1930
     K1=K                                                              QMA1940
     K=K2                                                              QMA1950
     IIN=0                                                             QMA1960
     ALPHA=1E75                                                        QMA1970
     SAS=G(ICAC+IOUT)                                                  QMA1980
     IF(SAS.GT.0.)ALPHA=ABS(G(IOUT))/SAS                               QMA1990
     IF(G(IOUT).LT.0.)GOTO898                                          QMA2000
     J=1                                                               QMA2010
     Z=X(LIOUT)-BL(LIOUT)                                              QMA2020
     GOTO899                                                           QMA2030
898  CONTINUE                                                          QMA2040
     J=0                                                               QMA2050
     Z=BU(LIOUT)-X(LIOUT)                                              QMA2060
899  CONTINUE                                                          QMA2070
     IF(Z.GE.ALPHA)GOTO25                                              QMA2080
     ALPHA=Z                                                           QMA2090
     LB=J                                                              QMA2100
     IIN=IOUT                                                          QMA2110
     LIIN=LIOUT                                                        QMA2120
     GOTO25                                                            QMA2130
 90  CONTINUE                                                          QMA2140
     K2=K1+1                                                           QMA2150
     IF( ABS(SIG-1.).LT.1.0E-30) GO TO 91                              QMA2170
     DO 901 I=K1,N                                                     QMA2180
901  G(IAS+I)=-G(IAS+I)                                                QMA2190
 91  CONTINUE                                                          QMA2200
     IF(IOUT.EQ.K1)GOTO97                                              QMA2210
     LT(IOUT)=LT(K1)                                                   QMA2220
     LT(K1)=LIOUT                                                      QMA2230
     G(IAS+IOUT)=G(IAS+K1)                                             QMA2240
     G(ICAC+IOUT)=G(ICAC+K1)                                           QMA2250
     G(ICAC+K1)=SAS                                                    QMA2260
     G(IOUT)=G(K1)                                                     QMA2270
     IF(K.EQ.0)GOTO93                                                  QMA2280
     DO 92 I=1,K                                                       QMA2290
     Z=A(K1,I)                                                         QMA2300
     A(K1,I)=A(IOUT,I)                                                 QMA2310
 92  A(IOUT,I)=Z                                                       QMA2320
 93  CONTINUE                                                          QMA2330
     IF(K2.EQ.IOUT)GOTO95                                              QMA2340
     I1=IOUT-1                                                         QMA2350
     DO 94 I=K2,I1                                                     QMA2360
 94  A(IOUT,I)=A(I,K1)                                                 QMA2370
 95  CONTINUE                                                          QMA2380
     IF(IOUT.EQ.N)GOTO97                                               QMA2390
     I1=IOUT+1                                                         QMA2400
     DO 96 I=I1,N                                                      QMA2410
 96  A(I,IOUT)=A(I,K1)                                                 QMA2420
 97  CONTINUE                                                          QMA2430
     G(K1)=0.                                                          QMA2440
     K=K1                                                              QMA2450
     IF(K.EQ.N)GOTO27
```

```
      DO 92 I=K2,N                                                      BQMA2460
      Z=G(IAS+I)/SAS                                                    BQMA2470
      A(I,K1)=Z                                                         BQMA2480
   92 G(ICAC+I)=G(ICAC+I)-Z*G(IAS+I)                                    BQMA2490
      K1=K2                                                             BQMA2500
      GOTO20                                                            BQMA2510
      END                                                               BQMA2520
      SUBROUTINE BQDMB(N,A,IA,G,K)                                      BQMB0010
      DIMENSION A(IA,1),G(1)                                            BQMB0030
      IF(K.EQ.0)RETURN                                                  BQMB0040
      ID=N+N                                                            BQMB0050
      G(N+1)=1./G(ID+1)                                                 BQMB0060
      IF(K.EQ.1)RETURN                                                  BQMB0070
      N1=K-1                                                            BQMB0080
      DO 111 I=1,N1                                                     BQMB0090
      I1=I+1                                                            BQMB0100
      A(I1,I)=-A(I1,I)                                                  BQMB0110
      IF(I.EQ.N1)GOTO102                                                BQMB0120
      II=I+2                                                            BQMB0130
      DO 101 J=II,K                                                     BQMB0140
      Z=A(J,I)                                                          BQMB0150
      J1=J-1                                                            BQMB0160
      DO 100 L=I1,J1                                                    BQMB0170
  100 Z=Z+A(J,L)*A(L,I)                                                 BQMB0180
  101 A(J,I)=-Z                                                         BQMB0190
  102 CONTINUE                                                          BQMB0200
      AA=1./G(ID+I1)                                                    BQMB0210
      G(N+I1)=AA                                                        BQMB0220
      DO 111 J=1,I                                                      BQMB0230
      Z=A(I1,J)*AA                                                      BQMB0240
      G(N+J)=G(N+J)+Z*A(I1,J)                                           BQMB0250
      IF(I.EQ.1)GOTO111                                                 BQMB0260
      J1=J+1                                                            BQMB0270
      DO 110 L=J1,I                                                     BQMB0280
  110 A(L,J)=A(L,J)+A(I1,L)*Z                                           BQMB0290
  111 A(I1,J)=Z                                                         BQMB0300
      RETURN                                                            BQMB0310
      END
      SUBROUTINE ALAGB(N,M,X)
      REAL X(N)
      COMMON/ALAGC/E,MM,KL,IS,MK,NU
      COMMON/ALAGD/G( 50)
      COMMON/ALAGE/C(150)
      COMMON/ALAGF/GG(25,50)
      COMMON/ALAGG/T(150)
      COMMON/SCALE/VSCAL(22),CSCAL(40),FSCAL
      COMMON/CONS/XM1,XM2,XM3,XM4,XM5,XM6,XM7,XM8,XM9,XM10,XM11,        SUM00030
     1XM12,XM13,XM14,XM15,XM16,XM17,XM18,XM19,XM20,XM21,XM22,XM23,XM24, SUM00035
     1XM25,XM26,XM27,XM28,PC,PI,EC,EC,FM,FC,VST,VSS,TSS,TSF,
     1  VD,TND,TFD,TRE,PE1,PE2,B   S1,BS2,BS3,VR,RCKCK,DI,              SUM0011
     2  DC,DK3,DK4,DK6,KS,KM,S,FR,RT                                   SUM00
      X1=X (1)    *VSCAL( 1)
      X2=X (2)    *VSCAL( 2)
      X3=X (3)    *VSCAL( 3)
      X4=X (4)    *VSCAL( 4)
```


ORIGINAL PAGE IS
OF POOR QUALITY

ALAG
BUILT-IN
SUBROUTINES

USER SUPPLIED
SUBROUTINE

```
      X5=X (5)    *VSCAL( 5)
      X6=X (6)    *VSCAL( 6)
      X7=X (7)    *VSCAL( 7)
      X8=X (8)    *VSCAL( 8)
      X9=X (9)    *VSCAL( 9)
      X10=X (10)  *VSCAL(10)
      X11=X (11)  *VSCAL(11)
      X12=X (12)  *VSCAL(12)
      X13=X (13)  *VSCAL(13)
      X14=X (14)  *VSCAL(14)
      X15=X (15)  *VSCAL(15)
      X16=X (16)  *VSCAL(16)
      X17=X (17)  *VSCAL(17)
      X18=X (18)  *VSCAL(18)
      X19=X (19)  *VSCAL(19)
      X20=X (20)  *VSCAL(20)
      X21=X (21)  *VSCAL(21)
      X22=X (22)  *VSCAL(22)
C     WRITE(6,8999)X1,X2,X3,X4,X17,X18,X19,X23
C8999 FORMAT('0TEST OF X VALU', 8F13.8)
 1000 Y1=DI*(X1*X1*X11+X4*X4*X12+X17*X17*X22)
      Y2=XM1*(X3*X3*X2*X2*X1+X6*X6*X5*X5*X4+X19*X19*X18*X18*X17)
      Y3=DK3*X9+DK4*X10+DK6*X21
      Y4=PO*(1./X8-1.)/KH+PO/X8/KS
      F=Y1+Y2+Y3+Y4
        F=F/FSCAL
      PIF=(XM4/X8)**2*(X13+X14)
      PQ=(XM4*XM8/X8)*(VST+.1*VBE)+XM10*(XM4*XM11/X8+XM9*XM12/X20)
      PD=XM7*VD/X6+XM15*(XM4*XM13/X8+XM9*XM14/X20)
      PCF=(XM4/X8)**2*X16+XM9*XM9*(X16+RCKCK*XM16/X21)/(3.*X20*X20)+(
     1RCKCK/X21)*(XM16*XM7**2+XM7*XM4/X8/X8-2.*XM7*XM7/X8)
     2+XM17*XM8*X22/X18/X18
    7 C(1)=PO*(1./X8-1.)-PIF-PQ-PD-PCF
    8 C(2)=X13*X3*X3-XM2*X1*X2*X2
   12 C(3)=X14*X6*X6-XM2*X5*X5*X4
    - C(4)=X9*X9+X15*X15*X9**3/X7-PE1**2*(X10*X10+(X15*X15*X9/X7)*
     1(X9-X10*XM18)**2)
      C(5)=X1*X1*X2*X2-X7*XM4/(X8*BS1)
   13 C(6)=X4*X4*X5*X5-X7*XM4/(X8*BS2*PE2)
      C(7)=XM6*X3*X2-X11*XM3+.5*X1
      C(8)=XM6*X6*X5-X12*XM3+.5*X4
    2 C(9)=X17*X17*X18*X18-(X20/BS5)*(XM4/X8+XM9/X20)
      C(10)=XM6*X19*X18-X22*XM3+.5*X17
      C(11) = VR-(XM4/X8 + XM9/X20)*RCKCK/X21/ED-XM7*XM8/2/ER/X21/ED
      C(12)=X16*X19*X19-XM2*X18*X18*X17
      C(13)=-1.0+XM25*XM27*X7*X10*X20*(XM28*X7/X15-1.0)
      C(14)=.969-X8
   11 C(15)=RT-X13-X14
      C(16)=X10-1.E-6
      C(17)=X9 - 1.E-6
C     WRITE(6,8994) (C(I), I=1,17)
C8994 FORMAT (E15.6)
      DO 103 I=1,22
  103 C(17+I)=X(I)
      G(1)=DI*X11*2*X1+XM1*X3*X3*X2*X2
```

CONSTRAINT EQUATIONS

DERIVATIVES OF THE OBJECTIVE FUNCTION

ORIGINAL PAGE IS OF POOR QUALITY

```
      G(2)=XM1*X3*X3*X1*2*X2
      G(3)=XM1*X2*X2*X1*2*X3
      G(4)=DI*X12*2*X4+XM1*X6*X6*X5*X5
      G(5)=XM1*X6*X6*X4*2*X5
      G(6)=XM1*X5*X5*X4*2*X6
      G(7)=0.0
      G(8)=-PO/X8/X8/KH-PO/KS/X8/X8
      G(9)=DK3
      G(10)=DK4
      G(11)=DI*X1*X1
      G(12)=DI*X4*X4
      G(17)=DI*X22*2*X17+XM1*X19*X19*X18*X18
      G(18)=XM1*X19*X19*X17*2*X18
      G(19)=XM1*X18*X18*X17*2*X19
      G(21)=DK6
      G(22)=DI*X17*X17
      GC(8,1)=-PC/X8/X8+(2./X8**3)*XM4*XM4*(X13+X14)+(XM4*XM8/X8**2)*(V
     1ST+.1*VBE)+XM10*(XM4*XM11/X8**2)
     2+XM7*VO/X8**2+XM15*(XM4*XM13/X8**2)+(2./X8**3)*XM4**2*X16
     3+2.*RCKCK*XM7*XM4/X8**3/X21-2.*RCKCK*XM7**2/X8/X8/X21
      GC(13,1)=-(XM4/X8)**2
      GC(14,1)=-(XM4/X8)**2
      GC(16,1)=-(XM4/X8)**2-XM9**2/(3.*X20**2)
      GC(18,1)=2*XM17*XM8*X22/X18/X18/X18
      GC(20,1)=XM10*XM9*XM12/X20**2+XM15*XM9*XM14/X20**2
     1+2.*XM9**2*(X16+RCKCK*XM16/X21)/(3.*X20**3)
      GC(21,1)=XM9**2*XM16*RCKCK/X21/X21/(3.*X20**2)+(RCKCK/X21/X21)*
     1(XM16*XM7**2+XM7*XM4/X8/X8-2.*XM7**2/X8)
      GC(22,1)=-XM17*XM8/X18/X18
      GC(13,2)=X3*X3
      GC(3,2)=X13*2*X3
      GC(2,2)=-XM2*X1*2*X2
      GC(1,2)=-XM2*X2*X2
      GC(14,3)=X6*X6
      GC(6,3)=X14*2*X6
      GC(5,3)=-XM2*X4*2*X5
      GC(4,3)=-XM2*X5*X5
      GC(15,4)=2.*X15*X9**3/X7-PE1*PE1*(2.*X15*X9/X7)*(X9-X10*XM18)**2
      GC(9,4)=2.*X9+3.*X9*X9*X15*X15/X7-PE1*PE1*((X15*X15/X7)*(X9-X10*
     1XM13)**2+(X15*X15*X9/X7)*2.*(X9-X10*XM18))
      GC(7,4)=-X15*X15*X9**3/X7/X7+PE1*PE1*(X15*X15*X9/X7/X7)*(X9-X10*
     1XM18)**2
      GC(10,4)=-PE1*PE1*(2.*X10-(X15*X15*X9/X7)*2.*(X9-X10*XM18)*XM18)
      GC(1,5)=X2*X2*2*X1
      GC(2,5)=X1*X1*2*X2
      GC(7,5)=-XM4/(X8*BS1)
      GC(8,5)=X7*XM4/(BS1*X8*X8)
      GC(4,6)=X5*X5*2*X4
      GC(5,6)=X4*X4*2*X5
      GC(7,6)=-XM4/(X8*BS2*PE2)
      GC(8,6)=X7*XM4/(PE2*BS2*X8*X8)
      GC(2,7)=XM6*X3
      GC(3,7)=XM6*X2
      GC(11,7)=-XM3
      GC(1,7)=.5
```

DERIVATIVES OF THE CONSTRAINTS

ORIGINAL PAGE IS
OF POOR QUALITY

```
      GC(5,8)=XM6*X6
      GC(6,8)=XM6*X5
      GC(12,8)=-XM3
      GC(4,8)=.5
      GC(17,9)=X19*X18*2*X17
      GC(18,9)=X17*X17*2*X18
      GC(20,9)=-(XM4/X8+XM9/X20)/BS5+(X20/BS5)*XM9/X20**2
      GC(8,9)=X20/BS5*XM4/X8**2
      GC(18,10)=XM6*X19
      GC(19,10)=XM6*X18
      GC(22,10)=-XM3
      GC(17,10)=.5
      GC(8,11)  =  (XM4/X8/X8)*RCKCK/X21/E0
      GC(20,11)  =  (XM9/X20/X20)*RCKCK/X21/E0
      GC(21,11)=(XM4/X8+XM9/X20)*RCKCK/X21/X21/EC+XM7*XM8/2/FR/X21/X21/E
     1C
      GC(16,12)=X19*X19
      GC(19,12)=X16*2*X19
      GC(18,12)=-XM2*X17*2*X18
      GC(17,12)=-XM2*X18*X18
      GC(7,13)=2.0*XM26*XM27*XM28*X7*X10*X20/X15-XM26*XM27*X10*X20
       GC(10,13)=XM26*XM27*X7*X20*(XM28*X7/X15-1.0)
      GC(15,13)=-XM26*XM27*XM28*X7**2*X10*X20/X15**2
       GC(20,13)=XM26*XM27*X7*X10*(XM28*X7/X15-1.0)
      GC(8,14)=-1.
      GC(13,15)=-1.
      GC(14,15)=-1.
      GC(10,16)=1.
      GC(9,17)=1.
      DO 104 I =1,22
104   GC(I,I+17)=1.0
      DO 101 I=1,N
      G(I)=G(I)*VSCAL(I)/FSCAL
101   CONTINUE
      DO 102 I=1,M
      C(I)=C(I)/CSCAL(I)
      DO 102   J=1,N
      GC(J,I)=GC(J,I)*VSCAL(J)/CSCAL(I)
102   CONTINUE
C
      WRITE(6,8995) (C(I), I=1,M)
C8995 FORMAT (4E15.6)
      RETURN
      END
```

USER SUPPLIED
SUBROUTINE

ORIGINAL PAGE IS
OF POOR QUALITY

APPENDIX F: DERIVATIONS OF CONSTRAINTS FOR BUCK-BOOST CONVERTER

Derivations of constraints which similar to the Boost Converter are omitted here. They can be derived using the same approach as illustrated in Appendix A.

F.1 Basic Relationships for Buck-Boost Converter

With reference to Fig. 1.3, one can derive the following basic relations

Input-output relationship

Transistor Q on:

$\Delta\phi(+)$ = flux increase in the energy storage inductor T

$$= \frac{E_i T_{on}}{N_p}$$

Transistor Q off:

$\Delta\phi(-)$ = flux decrease in the energy storage inductor T

$$= \frac{E_o T_{off}}{N_s}$$

When in stead state: $\Delta\phi(+) = \Delta\phi(-)$

$$\frac{E_0}{E_i} = \frac{nT_{on}}{T_{off}} \text{ , where } n = \frac{N_s}{N_p} \qquad (F.1)$$

Peak to Peak ripple Current 2d:

Transistor Q on:

current increment in the primary winding of T

$$\Delta i(t) = 2d = \frac{E_i E_o T}{L_p(E_o + nE_i)} \cdot \qquad (F.2)$$

Note: Derivations of transistor and diode loss constraints which are similar to the Boost converter are omitted here.

F.2 Output filter ESR Loss

Referring to the waveform of Fig. 6.2.

$$i_{c5}(t) = \begin{cases} -I_o, & 0 \leq t < T_{on} \\ \\ \dfrac{2d}{n(T_{on}-T)}\,(t-T_{on}) + I_i - I_o + \dfrac{d}{n}, & Ton \leq t \leq T \end{cases} \qquad (F.3)$$

$$\text{ESR loss} = \frac{1}{T}\int_0^T i_{c5}(t)\,[i_{c5}(t)\,R_5]\,dt$$

$$= \frac{1}{T}\int_0^{T_{on}} (-I_o)^2\,dt + \frac{1}{T}\int_{T_{on}}^{T} \left[\frac{2d}{n(T_{on}-T)}\,(t-T_{on}) + I_i - I_o + \frac{d}{n}\right]^2 dt$$

$$= \frac{T_{on}}{T} I_o^2 + \left(1 - \frac{T_{on}}{T}\right)\left[\frac{1}{3}\left(\frac{d}{n}\right)^2 + \left(\frac{I_i}{n} - I_o\right)^2\right] \qquad (F.4)$$

Substitute $\frac{T_{on}}{T} = \frac{E_o}{E_o + nE_i}$ $1 - \frac{T_{on}}{T} = \frac{nE_i}{E_o + nE_i}$,

$$I_i = \frac{P_o}{\eta_{eff} E_i} \frac{E_o + nE_i}{E_o}, \quad d = \frac{E_i E_o}{2L_p (E_o + nE_i) F}$$

into the above equation and we can get

$$\text{ESR loss} = \frac{E_o P_o^2}{(E_o + nE_i) E_o^2} + \frac{nE_i}{E_o + nE_i} \left[\frac{E_i^2 E_o^2}{12n^2 L_p^2 (E_o + nE_i)^2 F^2} + \left(\frac{(E_o + nE_i) P_o}{nE_o E_i \eta_{eff}} - \frac{P_o}{E_o} \right)^2 \right] R_5 \qquad (F.5)$$

F.3 Frequency dependent Source EMI Constraint

Referring to the $i_Q = \frac{2d}{T_{on}} t + (I_i - d), \quad 0 \leq t \leq T_{on}$ (F.6)

By Fourier series expansion

$$i_Q(t) = C_o + \sum_{k=1}^{\infty} A_k \cos k\omega_o t + \sum_{k=1}^{\infty} B_k \sin k\omega_o t \qquad (F.7)$$

where $A_k = \frac{2}{T} \int_0^{T_{on}} i_Q(t) \sin k\omega_o t \, dt$

$$B_k = \frac{2}{T} \int_0^{T_{on}} iQ(t) \sin k\omega_o t \, dt$$

A_1 = fundamental sine component

$$= \frac{2}{T} \int_0^{T_{on}} \left[\frac{2d}{T_{on}} + (I_i - d) \right] \cos \omega_o t \, dt$$

$$= \frac{2d}{\omega_o T} \sin \omega_o T_{on} + \frac{4d}{\omega_o^2 T_{on} T} \cos \omega_o T_{on} + \frac{2I_i \sin \omega_o T_{on}}{\omega_o T}$$

$$- \frac{4d}{\omega_o^2 T_{on} T} \qquad \text{(F.8)}$$

B_1 = fundamental cosine component

$$= \frac{2}{T} \int_0^{T_{on}} i_Q(t) \sin \omega_o t \, dt$$

$$= \frac{2}{T} \int_0^{T_{on}} \left[\frac{2d}{T_{on}} t + (I_i - d) \right] \sin \omega_o t \, dt$$

$$= \frac{-2d}{\omega_o T} \cos \omega_o T_{on} + \frac{4d}{\omega_o^2 T_{on} T} \sin \omega_o T_{on} - \frac{2I_i \cos \omega_o T_{on}}{\omega_o T}$$

$$+ \frac{2(I_i - d)}{\omega_o T} + \frac{4d}{\omega_o T} \qquad \text{(F.9)}$$

Now add the square of A_1 and B_1,

$$A_1^2 + B_1^2 = \left[\frac{2I_i}{\pi} \sin \frac{\omega_o T_{on}}{2} \right]^2$$

$$+ \left[\frac{2d}{\pi} \left(\cos \frac{\omega_o T_{on}}{2} - \frac{\sin \frac{\omega_o T_{on}}{2}}{\frac{\omega_o T_{on}}{2}} \right) \right]^2$$

let $A_1^2 + B_1^2 = A^2 + B^2$

where $A \triangleq \frac{2I_i}{\pi} \sin \frac{\omega_o T_{on}}{2}$

$$B \triangleq \frac{2d}{\pi} \left(\cos \frac{\omega_o T_{on}}{2} - \frac{\sin \frac{\omega_o T_{on}}{2}}{\frac{\omega_o T_{on}}{2}} \right) \qquad \text{(F.10)}$$

For Buck-Boost converter

$$I_i = \frac{P_o}{effE_i} \frac{(E_o + nE_i)}{E_o}$$

$$Ton = \frac{E_o T}{E_o + nE_i}$$

$$d = \frac{E_i E_o}{2L_p(E_o + nE_i)F} \qquad (F.11)$$

The EMI constraint becomes:

Required attenuation at switching frequency

$$= \frac{\text{EMI requirement}}{\text{Fundamental switching current}}$$

$$= \frac{\frac{S}{\sqrt{1+(\frac{F}{2000})}}}{\sqrt{A^2 + B^2}} \qquad (F.12)$$

APPENDIX G: Buck-Boost Converter Computer List

In this appendix, a complete computer list for the Buck-Boost converter is given. For the Buck-Boost converter the design unknown variable array is chosen as following:

$x(1) = \sqrt{A_1}$

$x(2) = \sqrt{N_1}$

$x(3) = \sqrt{A_{c1}}$

$x(4) = \sqrt{A_2}$

$x(5) = \sqrt{N_2}$

$x(6) = \sqrt{A_{c2}}$

$x(7) = L_1$

$x(8) = eff$

$x(9) = C_3$

$x(10) = C_4$

$x(11) = Z_1$

$x(12) = Z_2$

$x(13) = R_1$

$x(14) = R_2$

$x(15) = R_3$

$x(16) = R_p$

$x(17) = \sqrt{A_p}$

$x(18) = \sqrt{N_p}$

$x(19) = \sqrt{A_{cp}}$

$x(20) = L_p$

$x(21) = C_5$

$X(22) = Z_p$

ORIGINAL PAGE IS
OF POOR QUALITY

After the variable array is chosen, the user can start to simplify the constraints and chooses the constant terms of the constraints, and then take the first derivatives of all the cnstraints with respect to their own variables, then he can put all this information in computer code ready to use in the program. The constant terms, constraint equations and their first derivatives are shown as following:

G.1 Constant terms:

```
      XNEI=XN*EI
      PI=3.141592654
      XM1=4.*EC*EC
      XM2=4.*RC*EC
      XM3=1./(2.*PI)
      XM4=PO/EI
      XM5=SQRT(1./FW/PI)
      XM6=EO+XNEI
      XM7=PO/EO
      XM8=(EO+XNEI)/(EI*EO)
      XM9=EI*EO/(EO+XNEI)/FR/2.
      XM10=FR*((EC+VO)/XN+EI+2.*VST)/6.
      XM11=TSF+TSR
      XM12=TSF-TSR
      XM13=TNO+TFO+3.*TRE
      XM14=TNO-TFO-3.*TRE
      XM15=EO*PI/(EO+XNEI)
      XM16=EI/(EO+XNEI)/XN
      XM17=.176*SQRT(FR)
      XM18=1.+1./PE2
      XM19=FR/PE2
      XM20=(2.*PO*XM8/PI*SIN(XM15)*FR)**2
      XM21=((1/PI/XM8)*(COS(XM15)-SIN(XM15)/XM15))**2
      XM22= 1.+FR*FR/2000.**2
      XM23=(S*FR)**2
      XM24=S/(SQRT(1.+FR*FR/2000.**2)*EO/PI**2*SIN(PI*(EO-EI)/EI))
      XM25=FR*FR*XM24
      XM26=8.*PI**3*FR**3/PE2
      XM27=SQRT(1.+(FR/2000.)**2)/S
      RCKCK=RCK*CK
      XM28=4.*PO**2/PI**2*(SIN(XM15))**2*XM8**2
      XM29=4.*XM9**2/PI**2*(COS(XM15)-SIN(XM15)/XM15)**2
      XM30=4.*PI**2*FR**2
```

G.2 Constraint equations and objective function (F)

```
      Y1=DI*(X1*X1*X11+X4*X4*X12+X17*X17*X22)
      Y2=XM1*(X3*X3*X2*X2*X1+X6*X6*X5*X5*X4+X19*X19*X18*X18*X17)
      Y3=DK3*X9+DK4*X10+DK5*X21
      Y4=PO*(1./X8-1.)/KP+PO/KS/X8
      F=Y1+Y2+Y3+Y4
      PIF=(XM4/X8)**2*(X13+X14)
      PQ=(XM4/X8) *(VST+.1*VEE)+XM10*(XM8*PO*XM11/X8+XM9*XM12/X20)
      PD=XM7*VO/X8+(XM6*FR/12./XN)*(PO*XM8*XM13/X8+XM9*XM14/X20)
      PCF=EC/XM6*XM7**2*RCKCK/X21+(XM16*PCKCK/X21+X16)*(PO*PC*XM8**2/X8*
     1*2+XM9**2/3./X20**2)+XM7**2*(XN*EI/XM6-2./X8)*RCKCK/X21+1./XM8/X18
     2**2*XM17*X22
      C(1)=PC-(1./X8-1.)-PIF-PQ-PD-PCF
      C(2)=X13*X3*X3-XM2*X1*X2*X2
      C(3)=X14*X6*X6-XM2*X5*X5*X4
      C(4)=X9*X9+X15*X15*X9**3/X7-PE1**2*(X10*X10+(X15*X15*X9/X7)*
     1(X9-X10*XM18)**2)
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
C(5)=X1*X1*X2*X2-X7*XM4/(X8*BS1)
C(6)=X4*X4*X5*X5-X7*XM4/(X8*BS2*PE2)
C(7)=XM5*X3*X2-X11*XM3+.5*X1
C(8)=XM5*X6*X5-X12*XM3+.5*X4
C(9)=X17*X17*X18*X18-(X20/BSP)*(XM8*PO/X8+XM9/X20)
C(10)=XM5*X19*X18-X22*XM3+.5*X17
C(11) = VR-(PO*XM8/XN/X8+XM9/XN/X20)*RCKCK/X21/EO-PO/2./XM6/FR/X21
1/EO
C(12)=X16*X19*X19/2.-XM2*X18*X18*X17
C(13)=XM26*X7**2*X10/X15-XM30*X7*X10-XM27*(XM28/X8**2+XM29/X20**2)
1**0.5
C(14)=.97-X8
C(15)=RT-X13-X14
C(16)=X10-1.E-6
C(17)=X9-1.E-6
```

G.3 First derivatives of the constraints and the objective function

(1) Derivatives of objective function

```
G(1)=DI*X11*2.*X1+XM1*X3*X3*X2*X2
G(2)=XM1*X3*X3*X1*2.*X2
G(3)=XM1*X2*X2*X1*2.*X3
G(4)=DI*X12*2.*X4+XM1*X6*X6*X5*X5
G(5)=XM1*X6*X6*X4*2.*X5
G(6)=XM1*X5*X5*X4*2.*X6
G(8)=-PO/X8/X8/KH-PO/KS/X8**2
G(9)=DK3
G(10)=DK4
G(11)=DI*X1*X1
G(12)=DI*X4*X4
G(17)=DI*X22*2.*X17+XM1*X19*X19*X19*X18
G(18)=XM1*X19*X19*X17*2.*X18
G(19)=XM1*X18*X18*X17*2.*X19
G(21)=DK5
G(22) = DI*X17*X17
```

Where G(i) is the derivative of the objective function with respect to variable X(i)

(2) Derivatives of the constraints

In the following GC(j,i) means the derivatives of constraint C(i) with respect to variable x(j).

```
GC(8,1)=-PO/X8/X8+(2./X8**3)*XM4*XM4*(X13+X14)+(XM4/X8**2)*(V
1ST+.1*VBE)+XM10*  XM8*XM11*PO/X8**2
2+XM7*VC/X8**2+(XM6*FR/12./XN)*PO*XM8*XM13/X8**2
3+2.*PC**2*XM8**2/X9**3*(XM16*RCKCK/X21+X16)-2.*XM7**2*RCKCK/
4X21/X8**2
GC(13,1)=-(XM4/X8)**2
GC(14,1)=-(XM4/X8)**2
GC(16,1)=-PC*PC*XM8**2/X8**2-XM9**2/(3.*X20*X20)
GC(18,1)=2./XM8/X18**3*XM17*X22
GC(20,1)=XM10*XM9*XM12/X20**2+(XM16*RCKCK/X21+X16)*2.*(XM9**2/
1(3.*X20**3))+(XM6*FR/12./XN*XM9*XM14/X20**2)
GC(21,1)=EC/XM6*XM7**2*RCKCK/X21**2+XM16*RCKCK/X21**2*(PC*PC*XM8
1**2/X8**2+XM9*XM9/3./X20**2)+XM7**2*(XN*EI/XM6-2./X3)*PCKCK/X21**2
GC(22,1)=-1./XM9/X18**2*XM17
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      GC(13,2)=X3*X3
      GC(3,2)=X13*2.*X3
      GC(2,2)=-XM2*X1*2.*X2
      GC(1,2)=-XM2*X2*X2
C
      GC(14,3)=X6*X6
      GC(6,3)=X14*2.*X6
      GC(5,3)=-XM2 *X4*2.*X5
      GC(4,3)=-XM2*X5*X5
      GC(15,4)=2.*X15*X9**3/X7-PE1*PE1*(2.*X15*X9/X7)*(X9-X10*XM18)**2
      GC(9,4)=2.*X9+3.*X9*X9*X15*X15/X7-PE1*PE1*((X15*X15*X9/X7)*2.*(X9
     1-X10*XM18)+(X9-X10*XM18)**2*X15**2/X7)
      GC(7,4)=-X15*X15*X9**3/X7/X7+PE1*PE1*(X15*X15*X9/X7/X7)*(X9-X10*
     1XM18)**2
      GC(10,4)=-PE1*PE1*(2.*X10-(X15*X15*X9/X7)*2.*(X9-X10*XM18)*XM18)
      GC(1,5)=X2*X2*2.*X1
      GC(2,5)=X1*X1*2.*X2
      GC(7,5)=-XM4/(X8*BS1)
      GC(8,5)=X7*XM4/(BS1*X8*X8)
      GC(4,6)=X5*X5*2.*X4
      GC(5,6)=X4*X4*2.*X5
      GC(7,6)=-XM4/(X8*BS2*PE2)
      GC(8,6)=X7*XM4/(PE2*BS2*X8*X8)
C
      GC(2,7)=XM5*X3
      GC(3,7)=XM5*X2
      GC(11,7)=-XM3
      GC(1,7)=.5
      GC(5,8)=XM5*X6
      GC(6,8)=XM5*X5
C
      GC(12,8)=-XM3
      GC(4,8)=.5
      GC(17,9)=X18*X18*2.*X17
      GC(18,9)=X17*X17*2.*X18
      GC(20,9)=-(XM8*PO/X8+XM9/X20)/BSP+(X20/BSP)*XM9/X20**2
      GC(8,9)=X20/BSP*XM8*PO/X8**2
      GC(18,10)=XM5*X19
      GC(19,10)=XM5*X18
      GC(22,10)=-XM3
      GC(17,10)=.5
      GC(8,11) = PO*RCKCK* XM8/(XN*X8**2*X21)/EO
      GC(21,11) = (XM8*PO/X8+XM9/X20)/XN*RCKCK/X21**2/EO+PO/(2.*XM6*FR
     1*X21**2)/EO
      GC(20,11) = XM9*RCKCK/(XN*X20**2*X21)/EO
      GC(16,12)=X19*X19 *.5
      GC(19,12)=X16*X19
      GC(18,12)=-XM2*X17*2.*X18
      GC(17,12)=-XM2*X18*X18
      GC(7,13)=2.*XM26*X7*X10/X15-XM30*X10
      GC(8,13)=XM27*XM28/X9**3*SQRT(XM28/X8**2+XM29/X20**2)
      GC(10,13)=XM26*X7**2/X15-XM30*X7
      GC(15,13)=-XM26*X7**2*X10/X15**2
      GC(20,13)=XM27*XM29/X20**3/SQRT(XM28/X8**2+XM29/X20**2)
      GC(8,14)=-1.
      GC(13,15)=-1.
      GC(14,15)=-1.
      GC(10,16)=1.
      GC(9,17)=1.
```

With the above information available, the user are ready to combine it with the subroutines and ready to run his own program. The detailed computer program are provided in the following with some notes on it for the users to reference. The main program is out in the front and the user's supplied subroutine ALAGB is put at the end.

G.4 Computer List

A complete list of buck/boost converter optimization program is attached.

```
      REAL    L1,L2,L5,KC,DK3,DK4,DK5,KS,KH,S
      DIMENSION X(22)
      DIMENSION EPS(22)
      COMMON/CONS/XM1,XM2,XM3,XM4,XM5,XM6,XM7,XM8,XM9,XM10,XM11,          SUM00030
     1XM12,XM13,XM14,XM15,XM16,XM17,XM18,XM19,XM20,XM21,XM22,XM23,XM24,  SUM00035
     1XM25,XM26,XM27,XM28,XM29,XM30,PO,EI,EO,FC,FW,RO,VST,VBE,TSR,TSF,
     1 VD,TND,TFD,TRE,PE1,PE2,BS1,BS2,BSP,VR,RCKCK,DI,                    SUM0011
     2 DC,DK3,DK4,DK5,KS,KH,S,FR,RT,XN                                      SU
      NAMELIST/CON/PO,EI,EO,FC,FW,RO,VST,VBE,TSR,TSF,                    SUM00110
     1 VD,TND,TFD,TRE,PE1,PE2,B S1,BS2,BSP,VR,RCK,CK,DI,                  SUM0011
     2 DC,DK3,DK4,DK5,KS,KH,S,FR,RT,XN                                      SU
      NAMELIST/PARMET/ PIF,PO,PD,POF,PCAP,PMAG,PT,SIGMAP,WS,WH,
     1WI,WW,WC,WMAG,W
      NAMELIST/PARMS/N,M,K,MAXFN,IPR1,IPR2,IW,MODE,AKMIN,DFN
      NAMELIST/XIN/X
      COMMON/ALAGE/C(150)
      COMMON/ALAGF/GC(25,50)                                             MAIN0050
      COMMON/ALAGG/T(150)                                                MAIN0060
      COMMON/ALAGI/G2P(325)                                              MAIN0070
      COMMON/SCALE/VSCAL(22),CSCAL(39),FSCAL
      READ(5,10)N,M,K,MAXFN,IPR1,IPR2,IW,MODE                           MAIN0110
   10 FORMAT(16I5)                                                       MAIN0120
      READ(5,20)AK MIN,DFN                                               MAIN0130
      WRITE(6,PARMS)
      READ(5,CON)                                                        SUM00145
      WRITE(6,CON)                                                       SUM00150
      FSCAL=1.0
      CSCAL( 1)=1.0E+0
      CSCAL( 2)=1.0E-9
      CSCAL( 3)=1.0E-10
      CSCAL( 4)=1.0E-10
      CSCAL( 5)=1.0E-5
      CSCAL( 6)=1.0E-5
      CSCAL( 7)=1.0E-4
      CSCAL( 8)=1.0E-4
      CSCAL( 9)=1.0E-4
      CSCAL(10)=1.0E-3
      CSCAL(11)=1.0E-2
      CSCAL(12)=1.0E-9
      CSCAL(13)=1.0E-2
      CSCAL(14)=1.0E+0
      CSCAL(15)=1.0E-3
      CSCAL(16)=1.0E-1
      CSCAL(17)=1.0E-5
      DO 103 I=1,N
  103 CSCAL(I+17)=1.0
      VSCAL( 1)=1.0E-3
      VSCAL( 2)=1.0E 0
      VSCAL( 3)=1.0E-4
      VSCAL( 4)=1.0E-3
      VSCAL( 5)=1.0E 0
      VSCAL( 6)=1.0E-4
      VSCAL( 7)=1.0E-4
      VSCAL( 8)=1.0E-1
      VSCAL( 9)=1.0E-5
```


ORIGINAL PAGE IS
OF POOR QUALITY

```
      VSCAL(10)=1.0E-6
      VSCAL(11)=1.0E-2
      VSCAL(12)=1.0E-2
      VSCAL(13)=1.0E-2
      VSCAL(14)=1.0E-2
      VSCAL(15)=1.0E+0
      VSCAL(16)=1.0E-1
      VSCAL(17)=1.0E-3
      VSCAL(18)=1.0E0
      VSCAL(19)=1.0E-4
      VSCAL(20)=1.0E-5
      VSCAL(21)=1.0E-5
      VSCAL(22)=1.0E-2
      READ(5,XIN)                                                    SUM00185
      WRITE(6,XIN)
      DO  111  I=1,N
  111 EPS(I)=0.0001
   20 FORMAT(8F10.2)                                                 MAIN0160
      WRITE(6,8016) EPS(1)
 8016 FORMAT(F12.8 //)
      XNEI=XN*EI
      PI=3.141592654                                                 SUM00315
      XM1=4.*FC*DC
      XM2=4.*RO*FC

      XM3=1./(2.*PI)
      XM4=PO/EI
      XM5=SQRT(1./FW/PI)
      XM6=EO+XNEI
      XM7=PO/EO
      XM8=(EO+XNEI)/(EI*EO)
      XM9=EI*EO/(EO+XNEI)/FR/2.
      XM10=FR*((EO+VD)/XN+EI+2.*VST)/6.
      XM11=TSF+TSR
      XM12=TSF-TSR
      XM13=TND+TFD+3.*TRE
      XM14=TND-TFD-3.*TRE
      XM15=EO*PI/(EO+XNEI)
      XM16=EI/(EO+XNEI)/XN
      XM17=.176*SQRT(FR)
      XM18=1.+1./PE2
      XM19=FR/PE2
      XM20=(2.*PO*XM8/PI*SIN(XM15)*FR)**2
      XM21=((1/PI/XM8)*(COS(XM15)-SIN(XM15)/XM15))**2
      XM22= 1.+FR*FR/2000.**2
      XM23=(S*FR)**2
      XM24=S/(SQRT(1.+FR*FR/2000.**2)*EO/PI**2*SIN(PI*(EO-EI)/EI))
      XM25=FR*FR*XM24
      XM26=8.*PI**3*FR**3/PE2
      XM27=SQRT(1.+(FR/2000.)**2)/S
      RCKCK=RCK*CK
      XM28=4.*PO**2/PI**2*(SIN(XM15))**2*XM8**2
      XM29=4.*XM9**2/PI**2*(COS(XM15)-SIN(XM15)/XM15)**2
      XM30=4.*PI**2*FR**2
C     THIS CHECKS FOR SCALING
      X1 =X(1)
```


ORIGINAL PAGE IS
OF POOR QUALITY

```
      X2 =X(2)
      X3 =X(3)
      X4 =X(4)
      X5 =X(5)
      X6 =X(6)
      X7 =X(7)
      X8 =X(8)
      X9 =X(9)
      X10 =X(10)
      X11 =X(11)
      X12 =X(12)
      X13 =X(13)
      X14 =X(14)
      X15 =X(15)
      X16 =X(16)
      X17 =X(17)
      X18 =X(18)
      X19 =X(19)
      X20 =X(20)
      X21 =X(21)
      X22 =X(22)
1000  Y1=DI*(X1*X1*X11+X4*X4*X12+X17*X17*X22)
      Y2=XM1*(X3*X3*X2*X2*X1+X6*X6*X5*X5*X4+X19*X19*X18*X18*X17)
      Y3=DK3*X9+DK4*X10+DK5*X21
      Y4=PO*(1./X8-1.)/KH+PO/KS/X8
      F=Y1+Y2+Y3+Y4
      PIF=(XM4/X8)**2*(X13+X14)
      PQ=(XM4/X8) *(VST+.1*VBE)+XM10*(XM8*PO*XM11/X8+XM9*XM12/X20)
      PD=XM7*VD/X8+(XM6*FR/12./XN)*(PO*XM8*XM13/X8+XM9*XM14/X20)
      POF=EO/XM6*XM7**2*RCKCK/X21+(XM16*RCKCK/X21+X16)*(PO*PO*XM8**2/X8*
     1*2+XM9**2/3./X20**2)+XM7**2*(XN*EI/XM6-2./X8)*RCKCK/X21+1./XM8/X18
     2**2*XM17*X22
7     C(1)=PO*(1./X8-1.)-PIF-PQ-PD-POF
8     C(2)=X13*X3*X3-XM2*X1*X2*X2
12    C(3)=X14*X6*X6-XM2*X5*X5*X4
9     C(4)=X9*X9+X15*X15*X9**3/X7-PE1**2*(X10*X10+(X15*X15*X9/X7)*
     1(X9-X10*XM18)**2)
      C(5)=X1*X1*X2*X2-X7*XM4/(X8*BS1)
13    C(6)=X4*X4*X5*X5-X7*XM4/(X8*BS2*PE2)

      C(7)=XM5*X3*X2-X11*XM3+.5*X1
      C(8)=XM5*X6*X5-X12*XM3+.5*X4
2     C(9)=X17*X17*X18*X18-(X20/BSP)*(XM8*PO/X8+XM9/X20)
      C(10)=XM5*X19*X18-X22*XM3+.5*X17
      C(11) = VR-(PO*XM8/XN/X8+XM9/XN/X20)*RCKCK/X21/EO-PO/2./XM6/FR/X21
     1/EO
      C(12)=X16*X19*X19/2.-XM2*X18*X18*X17
      C(13)=XM26*X7**2*X10/X15-XM30*X7*X10-XM27*(XM28/X8**2+XM29/X20**2)
     1**0.5
      C(14)=.97-X8
11    C(15)=RT-X13-X14
      C(16)=X10-1.E-6
        C(17)=X9-1.E-6
         WRITE(6,1388)F
1388       FORMAT(F13.6)
        WRITE(6,10000)(C(I),I=1,17)
```


ORIGINAL PAGE IS
OF POOR QUALITY

```
10000          FORMAT(4X,5(E14.7,5X))
      DO 101 I=1,N
      X(I)=X(I)/VSCAL(I)
101         CONTINUE                                                    SUM00805
      WRITE(6,XIN)                                                      MAIN0170
      CALL ALAGA(N,M,K,X,EPS,AKMIN,DFN,MAXFN,IPR1,IPR2,IW,MODE)
      DO 102 I =1,N
102   X(I)=X(I)*VSCAL(I)
      WRITE(6,XIN)
      XL2=X(7)/PE2
      X1 =X(1)
      X2 =X(2)
      X3 =X(3)
      X4 =X(4)
      X5 =X(5)
      X6 =X(6)
      X7 =X(7)
      X8 =X(8)
      X9 =X(9)
      X10 =X(10)
      X11 =X(11)
      X12 =X(12)
      X13 =X(13)
      X14 =X(14)
      X15 =X(15)
      X16 =X(16)
      X17 =X(17)
      X18 =X(18)
      X19 =X(19)
      X20 =X(20)
      X21 =X(21)
      X22 =X(22)
      TX1=X(1)*X(1)
      TX2=X(2)*X(2)
      TX3=X(3)*X(3)
      TX4=X(4)*X(4)
      TX5=X(5)*X(5)
      TX6=X(6)*X(6)
      TX7=X(17)*X(17)
      TX8=X(18)*X(18)
      TX9=X(19)*X(19)/2
      WRITE(6,9025)TX1,TX2,TX3,TX4                                      SUM00925
 9025 FORMAT('0A1=',G15.5,' N1=',G15.5,' AC1=',G15.5,' A2=',G15.5)      SUM00930
      WRITE(6,9026)TX5,TX6,X(7),XL2
 9026 FORMAT('0N2=',G15.5,' AC2=',G15.5,' L1=',G15.5,' L2=',G15.5)      SUM00940
      WRITE(6,9027)X(9),X(10),X(11),X(12)                               SUM00945
 9027 FORMAT('0C3=',G15.5,' C4=',G15.5,' Z1=',G15.5,' Z2=',G15.5)       SUM00950
      WRITE(6,9028)X(13),X(14),X(15),X(16)                              SUM00955
 9028 FORMAT('0R1=',G15.5,' R2=',G15.5,' R3=',G15.5,' RP=',G15.5)       SUM00960
      WRITE(6,9001)TX7,TX8,TX9,X(20)                                    SUM00965
 9001 FORMAT('0AP=',G15.5,' NP=',G15.5,' ACP=',G15.5,' LP=',G15.5)      SUM00970
      WRITE(6,9002)X(21),X(22)                                          SUM00975
 9002 FORMAT('0C5=',G15.5,' ZP=',G15.5)                                 SUM00980
      WRITE(6,9003)FR,X(8)
 9003 FORMAT('0FR=',G15.5,' EFF=',G15.5)
```

```
      PIF=(XM4/X8)**2*(X13+X14)
      PO=(XM4/X8)*(VST+.1*VBE)+XM10*(XM8*PO*XM11/X8+XM9*XM12/X20)
      PD=XM7*VD/X8+(XM6*FR/12./XN)*(PO*XM8*XM13/X8+XM9*XM14/X20)
      PCAP=(EO/XM6)*XM7**2*RCKCK/X21+(XM16*RCKCK/X21)*(PO*PO*XM8*XM8
     1/X8**2+XM9**2/3./X20**2)+XM7**2*(XN*EI/XM6-2./X8)*RCKCK/X21
      POF=X16*(PO**2*XM8**2/X8**2+XM9**2/3./X20**2)+1./XM8/X18**2*XM17*X
     122
      PMAG=PIF+POF
      PT=PIF+PO+PD+POF+PCAP                                              SUM01020
      SIGMAP=PO/X8-PO
      RATIO=XL2/X(7)
      WS=PO/X(8)/KS                                                      SUM01085
      WH=PO*(1./X(8)-1.)/KH                                              SUM01090
      WI=DI*(X(1)*X(1)*X(11)+X(4)*X(4)*X(12)+X(17)*X(17)*X(22))
      WW=XM1*(X(3)*X(3)*X(2)*X(2)*X(1)+X(6)*X(6)*X(5)*X(5)*X(4)+X(19)*
     1X(19)*X(18)*X(18)*X(17))
      WC=DK3*X(9)+DK4*X(10)+DK5*X(21)
      WMAG=WI+WW
      W=WI+WW+WC+WS+WH                                                   SUM01115
        WRITE(6,XIN)
      WRITE(6,PARMET)
      WRITE(6,30)(X(I),I=1,N)                                            MAIN0180
   30 FORMAT('1',10X,'SOLUTION VECTOR',/,('0',10X,6E18.10))              MAIN0190
      STOP                                                               MAIN0210
      END                                                                MAIN0220
      BLOCK DATA
      COMMON/ALAGD/G(50)
      COMMON/ALAGF/GC(25,50)
      COMMON/ALAGI/G2P(325)                                               ALMAIN
      DATA G2P/325*0.0E0/
      DATA GC/1250*0.0E0/
      DATA G/50*0.0E0/
      END
      SUBROUTINE ALAGA(N,M,K,X,EPS,AKMIN,DFN,MAXFN,IPR1,IPR2,IW,MODE)    ALGA0010
      REAL X(1),EPS(1)                                                   ALGA0020
      COMMON/ALAGC/F,MM,KL,IS,MK,NU                                      ALGA0030
      COMMON/ALAGD/G(50)                                                 ALGA0040
      COMMON/ALAGE/C(150)                                                ALGA0050
      COMMON/ALAGF/GC(1250)                                              ALGA0060
      COMMON/ALAGG/T(150)                                                ALGA0070
      COMMON/ALAGH/GP(50)                                                ALGA0080
      COMMON/ALAGI/G2P(325)                                              ALGA0090
      COMMON/ALAGJ/V(50)                                                 ALGA0100
      COMMON/ALAGK/WW(150)                                               ALGA0110
      COMMON/ALAGL/W(2500)                                               ALGA0120
      COMMON/ALAGM/ZZ(100)                                               ALGA0130
      COMMON/ALAGN/LT(100)                                               ALGA0140
      EXTERNAL ALAGZ                                                     ALGA0150
 1000 FORMAT(30I4)                                                       ALGA0160
 1001 FORMAT(8E15.7)                                                     ALGA0170
      NU=MAX0(25,N)                                                      ALGA0180
      IF(M.GT.50)NU=N                                                    ALGA0190
      IX=N                                                               ALGA0200
      ICS=M                                                              ALGA0210
      ICB=N+M                                                            ALGA0220
```

```
      IS=M                                                              ALGA0230
      IL=IS+M                                                           ALGA0240
      IP=M                                                              ALGA0250
      ILT=M                                                             ALGA0260
      NN=N*(N+1)/2                                                      ALGA0270
      MM=M                                                              ALGA0280
      KL=K                                                              ALGA0290
      MINS=0                                                            ALGA0300
      AK=1E60                                                           ALGA0310
      R=1.                                                              ALGA0320
      MK=0                                                              ALGA0330
      DO 1 I=1,M                                                        ALGA0340
    1 C(ICS+I)=1.                                                       ALGA0350
      CALL ALAGB(N,M,X)                                                 ALGA0360
      WRITE(6,10000)(C(I),I=1,17)
10000      FORMAT(4X,5(E14.7,5X))
      DF=DFN                                                            ALGA0370
      IF(DFN.LT.0E0)DF=ABS(DFN*F)                                       ALGA0380
      IF(ABS(DFN).LT.1.0E-30) DF=F                                      ALGA0390
      IF(DF.LE.0.)DF=1.                                                 ALGA0400
      DO 2 I=1,M                                                        ALGA0410
      CC=C(I)                                                           ALGA0420
      IF(I.GT.K)CC=AMIN1(CC,0E0)                                        ALGA0430
      IF(ABS(CC).GT.C(ICS+I))C(ICS+I)=ABS(CC)                           ALGA0440
    2 CONTINUE                                                          ALGA0450
      IF(IPR1.EQ.0)GOTO4                                                ALGA0460
      IF(MOD(MINS,IPR1).NE.0)GOTO4                                      ALGA0470
      PRINT 1002                                                        ALGA0480
 1002 FORMAT('1ENTRY TO ALAGA'///'0CONSTRAINT SCALE PARAMETERS ARE')    ALGA0490
      PRINT 1001,(C(ICS+I),I=1,M)                                       ALGA0500
    4 CONTINUE                                                          ALGA0510
      IF(MODE.LT.0)GOTO5                                                ALGA0520
      DO 3 I=1,M                                                        ALGA0530
      T(IS+I)=2E0*DF/C(ICS+I)**2                                        ALGA0540
    3 T(I)=0.                                                           ALGA0550
    5 CONTINUE                                                          ALGA0560
      MD=IABS(MODE)                                                     ALGA0570
    8 CONTINUE                                                          ALGA0580
      MINS=MINS+1                                                       ALGA0590
      DO 9 I=1,NN                                                       ALGA0600
    9 W(I)=G2P(I)                                                       ALGA0610
      IF(IPR1.EQ.0)GOTO7                                                ALGA0620
      IF(MOD(MINS,IPR1).NE.0)GOTO7                                      ALGA0630
      PRINT 1003,MINS                                                   ALGA0640
 1003 FORMAT(////'0OUTER ITERATION NUMBER IS',I3)                       ALGA0650
      PRINT 1004                                                        ALGA0660
 1004 FORMAT('0X(I)')                                                   ALGA0670
      PRINT 1001,(X(I),I=1,N)                                           ALGA0680
      PRINT 1005                                                        ALGA0690
 1005 FORMAT('0THETA(I)')                                               ALGA0700
      PRINT 1001,(T(I),I=1,M)                                           ALGA0710
      PRINT 1006                                                        ALGA0720
 1006 FORMAT('0SIGMA(I)')                                               ALGA0730
      PRINT 1001,(T(IS+I),I=1,M)                                        ALGA0740
    7 CONTINUE                                                          ALGA0750
```

```
      CALL QNWTA(ALAGZ,N,X,PHI,GP,W,WW,DF,EPS,MD,MAXFN,IPR2,IEXIT)      ALGA0760
      CALL ALAGB(N,M,X)                                                 ALGA0770
      MD=3                                                              ALGA0780
      AKK=0.                                                            ALGA0790
      DO 10 I=1,M                                                       ALGA0800
      CC=C(I)                                                           ALGA0810
      IF(I.GT.K.AND.C(I).GE.T(I))CC=AMIN1(CC,0E0)                       ALGA0820
      T(I)=T(I)*T(IS+I)                                                 ALGA0830
      WW(I)=ABS(CC)/C(ICS+I)                                            ALGA0840
      IF(WW(I).GT.AKK)AKK=WW(I)                                         ALGA0850
   10 CONTINUE                                                          ALGA0860
      IF(IPR1.EQ.0)GOTO16                                               ALGA0870
      IF(MOD(MINS,IPR1).NE.0)GOTO16                                     ALGA0880
      PRINT 1007                                                        ALGA0890
 1007 FORMAT('0EXIT FROM QNWTA'/'0LAGRANGE MULTIPLIER ESTIMATES')       ALGA0900
      PRINT 1001,(T(I),I=1,M)                                           ALGA0910
      PRINT 1008                                                        ALGA0920
 1008 FORMAT('0LARGEST SCALED CONSTRAINT VIOLATION'/                    ALGA0930
     1' THIS ITERATION, BEST ITERATION')                                ALGA0940
      PRINT 1001,AKK,AK                                                 ALGA0950
      PRINT 1009                                                        ALGA0960
 1009 FORMAT('0CONSTRAINT RESIDUALS')                                   ALGA0970
      PRINT 1001,(C(I),I=1,M)                                           ALGA0980
      PRINT 1010                                                        ALGA0990
 1010 FORMAT('0SCALED CONSTRAINT VIOLATIONS')                           ALGA1000
      PRINT 1001,(WW(I),I=1,M)                                          ALGA1010
   16 CONTINUE                                                          ALGA1020
      IF(IEXIT.EQ.0  .OR.  IEXIT.EQ.3) GO TO 20
      IF(AKK.LE.AKMIN)GOTO20                                            ALGA1040
      IF(AKK.GE.AK)GOTO11                                               ALGA1050
      DO 15 I=1,NN                                                      ALGA1060
   15 G2P(I)=W(I)                                                       ALGA1070
      DO 17 I=1,M                                                       ALGA1080
      IF(I.GT.K.AND. ABS(T(I)     ).LT.1.0E-30.AND.C(I).GE.0E0) GO TO 17
      ZZ(IP+I)=-T(IS+I)*C(I)                                            ALGA1100
      IF(I.GT.K.AND.ZZ(IP+I).LT.-T(I))ZZ(IP+I)=-T(I)                    ALGA1120
   17 CONTINUE                                                          ALGA1130
      IF(MINS.EQ.1)GOTO40                                               ALGA1140
      GOTO18                                                            ALGA1150
   11 CONTINUE                                                          ALGA1160
      IF(ABS(AKK-AK).LT.1.0E-10) GO TO 101
      GO TO 105
  101 DO 102 I = 1,M
      IF(C(ICB+I).GE.4.0*WW(I)) GO TO 102
      DS=9E0*T(IS+I)
      T(IS+I)=1E1*T(IS+I)
      IF(IPR1.NE.0)PRINT 1011,I,T(IS+I)
      DO 103J=1,N
  103 V(J)=GC((I-1)*NU+J)
      CALL MULDA(G2P,N,V,DS,V,N,N,DS)
  102 CONTINUE
      GO TO 18
  105 CONTINUE
      DO 14 I=1,M                                                       ALGA1170
      IF(WW(I).LE.AK.OR.C(ICB+I).GE.4E0*WW(I))GOTO14                    ALGA1180
```


ORIGINAL PAGE IS
OF POOR QUALITY

```
      DS=9E0*T(IS+I)                                                    ALGA1190
      T(IS+I)=1E1*T(IS+I)                                               ALGA1200
      IF(IPR1.NE.0)PRINT 1011,I,T(IS+I)                                 ALGA1210
 1011 FORMAT('0SIGMA(',I3,')  INCREASED TO ',E15.7)                     ALGA1220
      DO 12 J=1,N                                                       ALGA1230
   12 V(J)=GC((I-1)*NU+J)                                               ALGA1240
      CALL MULDA(G2P,N,V,DS,V,N,N,DS)                                   ALGA1250
   14 CONTINUE                                                          ALGA1260
   18 CONTINUE                                                          ALGA1270
      DO 13 I=1,N                                                       ALGA1280
      IF(ABS(X(I)-G(IX+I)).GT.EPS(I))GOTO21                             ALGA1290
   13 CONTINUE                                                          ALGA1300
      PRINT 1013                                                        ALGA1310
 1013 FORMAT('0REQUESTED ACCURACY NOT OBTAINED')                        ALGA1320
   20 CONTINUE                                                          ALGA1330
      IF(IEXIT.EQ.0)PRINT 2000                                          ALGA1340
 2000 FORMAT('0MATRIX SET IN G2P BY USER IS NOT POSITIVE DEFINITE')     ALGA1350
      IF(IPR1.EQ.0)RETURN                                               ALGA1360
      PRINT 1012                                                        ALGA1370
 1012 FORMAT('0BEST SOLUTION OBTAINED'/'0F,(G(I),I=1,N)')               ALGA1380
      PRINT 1001,F,(G(I),I=1,N)                                         ALGA1390
      RETURN                                                            ALGA1400
   21 CONTINUE                                                          ALGA1410
      IF(AKK.LT.AK)GOTO40                                               ALGA1420
      DO 32 I=1,M                                                       ALGA1430
   32 V(I)=T(IL+I)                                                      ALGA1440
      GOTO70                                                            ALGA1450
   40 CONTINUE                                                          ALGA1460
      MK=0                                                              ALGA1470
      KK=0                                                              ALGA1480
      DO 41 I=1,M                                                       ALGA1490
      T(IL+I)=T(I)                                                      ALGA1500
      C(ICB+I)=WW(I)                                                    ALGA1510
      IF(I.GT.K.AND. ABS(T(IL+I)  ).LT.1.0E-30.AND.C(I).GE.0E0) GO TO 41
      KK=KK+1                                                           ALGA1530
      LT(ILT+KK)=I                                                      ALGA1540
      GP(KK)=-1E30                                                      ALGA1550
      IF(I.GT.K)GP(KK)=-T(IL+I)                                         ALGA1560
      V(KK)=1E30                                                        ALGA1570
      ZZ(KK)=-C(I)                                                      ALGA1580
   41 CONTINUE                                                          ALGA1590
      IF(KK.EQ.0)GOTO20                                                 ALGA1600
      DO 42 I=1,N                                                       ALGA1610
   42 G(IX+I)=X(I)                                                      ALGA1620
      KKK=KK*(KK+1)/2                                                   ALGA1630
      II=MAX0(KKK+NN,KK*KK)                                             ALGA1640
      IF(II.LE.IW)GOTO50                                                ALGA1650
      PRINT 2001,II                                                     ALGA1660
 2001 FORMAT('0INCREASE STORAGE IN COMMON/ALAGL TO',I7,'ELEMENTS')      ALGA1670
      RETURN                                                            ALGA1680
   50 CONTINUE                                                          ALGA1690
      II=IW-KKK                                                         ALGA1700
      DO 53 I=1,KK                                                      ALGA1710
      LI=LT(ILT+I)                                                      ALGA1720
      DO 51 JJ=1,N                                                      ALGA1730
```


ORIGINAL PAGE IS
OF POOR QUALITY

```
   51 X(JJ)=GC((LI-1)*NU+JJ)                                                    ALGA1740
      CALL MULDE(W,N,X,X,N)                                                     ALGA1750
      DO 53 J=1,I                                                               ALGA1760
      LJ=LT(ILT+J)                                                              ALGA1770
      Z=0.                                                                      ALGA1780
      DO 52 JJ=1,N                                                              ALGA1790
   52 Z=Z+X(JJ)*GC((LJ-1)*NU+JJ)                                                ALGA1800
      II=II+1                                                                   ALGA1810
   53 W(II)=Z                                                                   ALGA1820
      JJ=IW-KKK                                                                 ALGA1830
      II=0                                                                      ALGA1840
      DO 56 I=1,KK                                                              ALGA1850
      DO 55 J=1,I                                                               ALGA1860
      JJ=JJ+1                                                                   ALGA1870
   55 W(II+J)=W(JJ)                                                             ALGA1880
   56 II=II+KK                                                                  ALGA1890
      CALL BQDMA(KK,W,KK,ZZ,GP,V,T,Z,LT,JJ,WW)                                  ALGA1900
      IF(IPR1.EQ.0)GOTO59                                                       ALGA1910
      IF(MOD(MINS,IPR1).NE.0)GOTO59                                             ALGA1920
      PRINT 1020,KK                                                             ALGA1930
 1020 FORMAT(I4,' ACTIVE CONSIRAINTS, NUMBERED')                                ALGA1940
      PRINT 1000,(LT(ILT+I),I=1,KK)                                             ALGA1950
      PRINT 1021                                                                ALGA1960
 1021 FORMAT('0LAGRANGE MULTIPLIER CORRECTIONS FOR ACTIVE CONSTRAINTS')         ALGA1970
      PRINT 1001,(T(I),I=1,KK)                                                  ALGA1980
   59 CONTINUE                                                                  ALGA1990
      DO 60 I=1,M                                                               ALGA2000
   60 V(I)=T(IL+I)                                                              ALGA2010
      DO 62 I=1,KK                                                              ALGA2020
      LI=LT(ILT+I)                                                              ALGA2030
      V(LI)=V(LI)+T(I)                                                          ALGA2040
      IF(ABS(ZZ(IP+LI)).LE.1.0E-30) GO TO 62
      Z=4E0*ABS((T(I)-ZZ(IP+LI))/ZZ(IP+LI))                                     ALGA2050
      IF(Z.LE.1E0)GOTO62                                                        ALGA2060
      DS=(Z-1E0)*T(IS+LI)                                                       ALGA2070
      T(IS+LI)=Z*T(IS+LI)                                                       ALGA2080
      IF(IPR1.NE.0)PRINT 1011,LI,T(IS+LI)                                       ALGA2090
      DO 61 J=1,N                                                               ALGA2100
   61 GP(J)=GC((LI-1)*NU+J)                                                     ALGA2110
      CALL MULDA(G2P,N,GP,DS,GP,N,N,DS)                                         ALGA2120
   62 CONTINUE                                                                  ALGA2130
      AK=AKK                                                                    ALGA2140
   70 CONTINUE                                                                  ALGA2150
      DO 71 I=1,M                                                               ALGA2160
   71 T(I)=V(I)/T(IS+I)                                                         ALGA2170
      DO 72 I=1,N                                                               ALGA2180
   72 X(I)=G(IX+I)                                                              ALGA2190
      DF=1E50                                                                   ALGA2200
      GOTO8                                                                     ALGA2210
      END                                                                       ALGA2220
      SUBROUTINE ALAGZ(N,X,PHI,GPHI)                                            ALGZ0010
      REAL X(1),GPHI(1)                                                         ALGZ0020
      COMMON/ALAGC/F,M,K,IS,MK,NU                                               ALGZ0030
      COMMON/ALAGD/G(50)                                                        ALGZ0040
      COMMON/ALAGE/C(150)                                                       ALGZ0050
```

```
      COMMON/ALAGF/GC(1250)                                              ALGZ0060
      COMMON/ALAGG/T(150)                                                ALGZ0070
      IF(MK.EQ.1)CALL ALAGB (N,M,X)                                      ALGZ0080
      MK=1                                                               ALGZ0090
      PHI=0.                                                             ALGZ0100
      DO 10 I=1,N                                                        ALGZ0110
   10 GPHI(I)=G(I)                                                       ALGZ0120
      DO 12 I=1,M                                                        ALGZ0130
      CC=C(I)-T(I)                                                       ALGZ0140
      IF(I.GT.K)CC=AMIN1(CC,0E0)                                         ALGZ0150
      Y=T(IS+I)*CC                                                       ALGZ0160
      IF( ABS(Y).LT.1.0E-30) GO TO 12
      PHI=PHI+Y*CC                                                       ALGZ0180
      DO 11 J=1,N                                                        ALGZ0190
   11 GPHI(J)=GPHI(J)+Y*GC((I-1)*NU+J).                                  ALGZ0200
   12 CONTINUE                                                           ALGZ0210
      PHI=.5E0*PHI+F                                                     ALGZ0220
      RETURN                                                             ALGA0230
      END                                                                ALGZ0240
      SUBROUTINE QNWTA(FUNCT,N,X,F,G,H,W,DFN,EPS,MODE,MAXFN,IPRINT,      QNTA0010
     1IEXIT)                                                             QNTA0020
      REAL X(1),G(1),H(1),W(1),EPS(1)                                    QNTA0030
      CALL FUNCT(N,X,F,G)
      IF(IPRINT.NE.0)PRINT 1000                                          QNTA0040
 1000 FORMAT('1ENTRY TO QNWTA'/)                                         QNTA0050
      NN=N*(N+1)/2                                                       QNTA0060
      IG=N                                                               QNTA0070
      IGG=N+N                                                            QNTA0080
      IS=IGG                                                             QNTA0090
      IEXIT=0                                                            QNTA0100
      IR=N                                                               QNTA0110
      IF(MODE.EQ.3)GOTO15                                                QNTA0120
      IF(MODE.EQ.2)GOTO10                                                QNTA0130
      IJ=NN+1                                                            QNTA0140
      DO 5 I=1,N                                                         QNTA0150
      DO 6 J=1,I                                                         QNTA0160
      IJ=IJ-1                                                            QNTA0170
    6 H(IJ)=0.                                                           QNTA0180
    5 H(IJ)=1.                                                           QNTA0190
      GOTO15                                                             QNTA0200
   10 CONTINUE                                                           QNTA0210
      CALL MULDB(H,N,IR)                                                 QNTA0220
      IF(IR.LT.N)RETURN                                                  QNTA0230
   15 CONTINUE                                                           QNTA0240
      Z=F                                                                QNTA0250
      ITN=0                                                              QNTA0260
      CALL FUNCT(N,X,F,G)                                                QNTA0270
      IFN=1                                                              QNTA0280
      DF=DFN                                                             QNTA0290
      IF(ABS(DFN).LT.1.0E-30) DF=F-Z                                     QNTA0300
      IF(DFN.LT.0.)DF=ABS(DF*F)                                          QNTA0310
      IF(DF.LE.0.)DF=1.                                                  QNTA0320
   20 CONTINUE                                                           QNTA0330
      IF(IPRINT.EQ.0)GOTO21                                              QNTA0340
      IF(MOD(ITN,IPRINT).NE.0)GOTO21                                     QNTA0350
```

```
      PRINT 1001,ITN,IFN                                              QNTA0360
 1001 FORMAT(24I5)                                                    QNTA0370
      PRINT 1002,F                                                    QNTA0380
 1002 FORMAT((8E15.7))                                                QNTA0390
      IF(IPRINT.LT.0)GOTO21                                           QNTA0400
      PRINT 1002,(X(I),I=1,N)                                         QNTA0410
      PRINT 1002,(G(I),I=1,N)                                         QNTA0420
   21 CONTINUE                                                        QNTA0430
      ITN=ITN+1                                                       QNTA0440
      DO 22 I=1,N                                                     QNTA0450
   22 W(IG+I)=G(I)                                                    QNTA0460
      CALL MULDE(H,N,G,W,IR)                                          QNTA0470
      GS=0.                                                           QNTA0480
      DO 29 I=1,N                                                     QNTA0490
      W(IS+I)=-G(I)                                                   QNTA0500
   29 GS=GS-G(I)*W(IG+I)                                              QNTA0510
      IEXIT=2                                                         QNTA0520
      IF(GS.GE.0.)GOTO92                                              QNTA0530
      GS0=GS                                                          QNTA0540
      ALPHA=-2.*DF/GS                                                 QNTA0550
      IF(ALPHA.GT.1.)ALPHA=1.                                         QNTA0560
      DF=F                                                            QNTA0570
      TOT=0.                                                          QNTA0580
   30 CONTINUE                                                        QNTA0590
C     IEXIT=3                                                         QNTA0600
C     IF(IFN.EQ.MAXFN)GOTO92                                          QNTA0610
      IEXIT=4
      IF(IFN.GT.MAXFN) GO TO 92                                       QNTA0620
      ICON=0                                                          QNTA0630
      IEXIT=1                                                         QNTA0640
      DO 31 I=1,N                                                     QNTA0650
      Z=ALPHA*W(IS+I)                                                 QNTA0660
      IF(ABS(Z).GE.EPS(I))ICON=1                                      QNTA0670
   31 X(I)=X(I)+Z                                                     QNTA0680
      CALL FUNCT(N,X,FY,G)                                            QNTA0690
      IFN=IFN+1                                                       QNTA0700
      GYS=0.                                                          QNTA0710
      DO 32 I=1,N                                                     QNTA0720
   32 GYS=GYS+G(I)*W(IS+I)                                            QNTA0730
      IF(FY.GE.F)GOTO40                                               QNTA0740
      IF(ABS(GYS/GS0).LE..9)GOTO50                                    QNTA0750
      IF(GYS.GT.0.)GOTO40                                             QNTA0760
      TOT=TOT+ALPHA                                                   QNTA0770
      Z=10.                                                           QNTA0780
      IF(GS.LT.GYS)Z=GYS/(GS-GYS)                                     QNTA0790
      IF(Z.GT.10.)Z=10.                                               QNTA0800
      ALPHA=ALPHA*Z                                                   QNTA0810
      F=FY                                                            QNTA0820
      GS=GYS                                                          QNTA0830
      GOTO30                                                          QNTA0840
   40 CONTINUE                                                        QNTA0850
      DO 41 I=1,N                                                     QNTA0860
   41 X(I)=X(I)-ALPHA*W(IS+I)                                         QNTA0870
      IF(ICON.EQ.0)GOTO92                                             QNTA0880
      Z=3.*(F-FY)/ALPHA+GYS+GS                                        QNTA0890
```

```
      IF(ABS(Z).GT.1.0E35) GO TO 36                                    ONTA0890
      GO TO 37                                                         ONTA0900
36    Z=1.0
      GO TO 39
37    CONTINUE
      ZZ=SQRT(Z**2-GS*GYS)                                             ONTA0910
      Z=1.-(GYS+ZZ-Z)/(2.*ZZ+GYS-GS)                                   ONTA0920
39    CONTINUE
      ALPHA=ALPHA*Z                                                    ONTA0930
      GOTO30                                                           ONTA0940
   50 CONTINUE
      ALPHA=TOT+ALPHA                                                  ONTA0950
      F=FY                                                             ONTA0960
      IF(ICON.EQ.0)GOTO90                                              ONTA0970
      DF=DF-F                                                          ONTA0980
      DGS=GYS-GS0                                                      ONTA0990
      DO 51 I=1,N                                                      ONTA1000
      W(IGG+I)=G(I)                                                    ONTA1010
   51 G(I)=-W(IG+I)                                                    ONTA1020
      IF(DGS+ALPHA*GS0.GT.0.)GOTO60                                    ONTA1030
C    COMPLEMENTARY DFP FORMULA                                         ONTA1040
      SIG=1./GS0                                                       ONTA1050
      IR=-IR                                                           ONTA1060
      CALL MULDA(H,N,G,SIG,W,IR,1,0.)                                  ONTA1070
      DO 52 I=1,N                                                      ONTA1080
   52 G(I)=W(IGG+I)-W(IG+I)                                            ONTA1090
      SIG=1./(ALPHA*DGS)                                               ONTA1100
      IR=-IR                                                           ONTA1110
      CALL MULDA(H,N,G,SIG,W,IR,0,0.)                                  ONTA1120
      GOTO70                                                           ONTA1130
   60 CONTINUE
C     DFP FORMULA                                                     ONTA1140
      ZZ=ALPHA/(DGS-ALPHA*GS0)                                         ONTA1150
      SIG=-ZZ                                                          ONTA1160
      CALL MULDA(H,N,G,SIG,W,IR,1,1E-7)                                ONTA1170
      Z=DGS*ZZ-1.                                                      ONTA1180
      DO 61 I=1,N                                                      ONTA1190
   61 G(I)=W(IGG+I)+Z*W(IG+I)                                          ONTA1200
      SIG=1./(ZZ*DGS**2)                                               ONTA1210
      CALL MULDA(H,N,G,SIG,W,IR,0,0.)                                  ONTA1220
   70 CONTINUE                                                         ONTA1230
      DO 71 I=1,N                                                      ONTA1240
   71 G(I)=W(IGG+I)                                                    ONTA1250
      GOTO20                                                           ONTA1260
   92 CONTINUE                                                         ONTA1270
      DO 91 I=1,N                                                      ONTA1280
   91 G(I)=W(IG+I)                                                     ONTA1290
   90 CONTINUE                                                         ONTA1300
      IF(IPRINT.EQ.0)RETURN                                            ONTA1310
      PRINT 1001,ITN,IFN,IEXIT                                         ONTA1320
      PRINT 1002,F                                                     ONTA1330
      PRINT 1002,(X(I),I=1,N)                                          ONTA1340
      PRINT 1002,(G(I),I=1,N)                                          ONTA1350
      RETURN                                                           ONTA1360
      END                                                              ONTA1370
```

```
      SUBROUTINE MULDA(A,N,Z,SIG,W,IR,MK,EPS)                          MUDA0010
      DI MENSION A(1),Z(1),W(1)                                        MUDE0020
C     UPDATE FACTORS GIVEN IN A BY    SIG*Z*ZTRANSPOSE                 MUDA0030
      IF(N.GT.1)GOTO1                                                  MUDA0040
      A(1)=A(1)+SIG *Z(1)**2                                           MUDA0050
      IR=1                                                             MUDA0060
      IF(A(1).GT.0.)RETURN                                             MUDA0070
      A(1)=0.                                                          MUDA0080
      IR=0                                                             MUDA0090
      RETURN                                                           MUDA0100
    1 CONTINUE                                                         MUDA0110
      NP=N+1                                                           MUDA0120
      IF(SIG.GT.0.)GOTO40                                              MUDA0130
      IF( ABS(SIG).LT.1.0E-30 .OR.IR.EQ.0) RETURN                      MUDA0150
      TI=1./SIG                                                        MUDA0160
      IJ=1                                                             MUDA0170
      IF(MK.EQ.0)GOTO10                                                MUDA0180
      DO 7 I=1,N                                                       MUDA0200
      IF( ABS(A(IJ)).GT.1.0E-30) TI=TI+W(I)**2/A(IJ)                   MUDA0210
    7 IJ=IJ+NP-I                                                       MUDA0220
      GOTO20                                                           MUDA0230
   10 CONTINUE                                                         MUDA0240
      DO 11 I=1,N                                                      MUDA0250
   11 W(I)=Z(I)                                                        MUDA0260
      DO 15 I=1,N                                                      MUDA0270
      IP=I+1                                                           MUDA0280
      V=W(I)                                                           MUDA0290
      IF(A(IJ).GT.0.)GOTO12                                            MUDA0300
      W(I)=0.                                                          MUDA0310
      IJ=IJ+NP-I                                                       MUDA0320
      GOTO15                                                           MUDA0330
   12 CONTINUE                                                         MUDA0340
      TI=TI+V**2/A(IJ)                                                 MUDA0350
      IF(I.EQ.N)GOTO14                                                 MUDA0360
      DO 13 J=IP,N                                                     MUDA0370
      IJ=IJ+1                                                          MUDA0380
   13 W(J)=W(J)-V*A(IJ)                                                MUDA0390
   14 IJ=IJ+1                                                          MUDA0400
   15 CONTINUE                                                         MUDA0410
   20 CONTINUE                                                         MUDA0420
      IF(IR.LE.0 )GOTO21                                               MUDA0430
      IF(TI.GT.0.)GOTO22                                               MUDA0440
      IF(MK-1)40,40,23                                                 MUDA0450
   21 TI=0.                                                            MUDA0460
      IR=-IR-1                                                         MUDA0470
      GOTO23                                                           MUDA0490
   22 TI=EPS/SIG                                                       MUDA0500
      IF( ABS(EPS).LT.1.0E-30) IR=IR-1                                 MUDA0510
   23 CONTINUE                                                         MUDA0520
      MM=1                                                             MUDA0540
      TIM=TI
      DO 30 I=1,N
      J=NP-I
      IJ=IJ-I
      IF( ABS(A(IJ)).GT.1.0E-30)TIM=TI-W(J)**2/A(IJ)
```

```
      W(J)=TI                                                          MUDA0560
   30 TI=TIM                                                           MUDA0570
      GOTO41                                                           MUDA0580
   40 CONTINUE                                                         MUDA0590
      MM=0                                                             MUDA0600
      TIM=1./SIG                                                       MUDA0610
   41 CONTINUE                                                         MUDA0620
      IJ=1                                                             MUDA0630
      DO 66 I=1,N                                                      MUDA0640
      IP=I+1                                                           MUDA0650
      V=Z(I)                                                           MUDA0660
      IF(A(IJ).GT.0.)GOTO53                                            MUDA0670
      IF(IR.GT.0 .OR.SIG.LT.0..OR. ABS(V).LT.1.0E-30) GO TO 52
      IR=1-IR                                                          MUDA0690
      A(IJ)=V**2/TIM                                                   MUDA0700
      IF(I.EQ.N)RETURN                                                 MUDA0710
      DO 51 J=IP,N                                                     MUDA0720
      IJ=IJ+1                                                          MUDA0730
   51 A(IJ)=Z(J)/V                                                     MUDA0740
      RETURN                                                           MUDA0750
   52 CONTINUE                                                         MUDA0760
      TI=TIM                                                           MUDA0770
      IJ=IJ+NP-I                                                       MUDA0780
      GOTO66                                                           MUDA0790
   53 CONTINUE                                                         MUDA0800
      AL=V/A(IJ)                                                       MUDA0810
      IF(MM)54,54,55                                                   MUDA0820
   54 TI=TIM+V*AL                                                      MUDA0830
      GOTO56                                                           MUDA0840
   55 TI=W(I)                                                          MUDA0850
   56 CONTINUE                                                         MUDA0860
      R=TI/TIM                                                         MUDA0870
      A(IJ)=A(IJ)*R                                                    MUDA0880
      IF( ABS(R).LT.1.0E-30) GO TO 70
      IF(I.EQ.N)GOTO70                                                 MUDA0900
      B=AL/TI                                                          MUDA0910
      IF(R.GT.4.)GOTO62                                                MUDA0920
      DO 61 J=IP,N                                                     MUDA0930
      IJ=IJ+1                                                          MUDA0940
      Z(J)=Z(J)-V*A(IJ)                                                MUDA0950
   61 A(IJ)=A(IJ)+B*Z(J)                                               MUDA0960
      GOTO64                                                           MUDA0970
   62 GM=TIM/TI                                                        MUDA0980
      DO 63 J=IP,N                                                     MUDA0990
      IJ=IJ+1                                                          MUDA1000
      Y=A(IJ)                                                          MUDA1010
      A(IJ)=B*Z(J)+Y*GM                                                MUDA1020
   63 Z(J)=Z(J)-V*Y                                                    MUDA1030
   64 CONTINUE                                                         MUDA1040
      TIM=TI                                                           MUDA1050
      IJ=IJ+1                                                          MUDA1060
   66 CONTINUE                                                         MUDA1070
   70 CONTINUE                                                         MUDA1080
      IF(IR.LT.0)IR=-IR                                                MUDA1090
      RETURN                                                           MUDA1100
```


ORIGINAL PAGE IS
OF POOR QUALITY

```
      END                                                               MUDA1110
      SUBROUTINE  MULDB(A,N,IR)                                         MUDB0010
      DI MENSION A(1)                                                   MUDB0020
      IR=N                                                              MUDB0030
      IF(N.GT.1)GOTO100                                                 MUDB0040
      IF(A(1).GT.0.)RETURN                                              MUDB0050
      A(1)=0.                                                           MUDB0060
      IR=0                                                              MUDB0070
      RETURN                                                            MUDB0080
  100 CONTINUE                                                          MUDB0090
      NP=N+1                                                            MUDB0100
      II=1                                                              MUDB0110
      DO 104 I=2,N                                                      MUDB0120
      AA=A(II)                                                          MUDB0130
      NI=II+NP-I                                                        MUDB0140
      IF(AA.GT.0.)GOTO101                                               MUDB0150
      A(II)=0.                                                          MUDB0160
      IR=IR-1                                                           MUDB0170
      II=NI+1                                                           MUDB0180
      GOTO104                                                           MUDB0190
  101 CONTINUE                                                          MUDB0200
      IP=II+1                                                           MUDB0210
      II=NI+1                                                           MUDB0220
      JK=II                                                             MUDB0230
      DO 103 IJ=IP,NI                                                   MUDB0240
      V=A(IJ)/AA                                                        MUDB0250
      DO 102 IK=IJ,NI                                                   MUDB0260
      A(JK)=A(JK)-A(IK)*V                                               MUDB0270
  102 JK=JK+1                                                           MUDB0280
  103 A(IJ)=V                                                           MUDB0290
  104 CONTINUE                                                          MUDB0300
      IF(A(II).GT.0.)RETURN                                             MUDB0310
      A(II)=0.                                                          MUDB0320
      IR=IR-1                                                           MUDB0330
      RETURN                                                            MUDB0340
      END                                                               MUDB0350
      SUBROUTINE  MULDE(A,N,Z,W,IR)                                     MUDE0010
      DIMENSION A(1),Z(1),W(1)                                          MUDA0020
C   MULTIPLY A VECTOR Z BY THE INVERSE OF THE FACTORS GIVEN IN A        MUDE0030
      IF(IR.LT.N)RETURN                                                 MUDE0040
      W(1)=Z(1)                                                         MUDE0050
      IF(N.GT.1)GOTO400                                                 MUDE0060
      Z(1)=Z(1)/A(1)                                                    MUDE0070
      RETURN                                                            MUDE0080
  400 CONTINUE                                                          MUDE0090
      DO 402 I=2,N                                                      MUDE0100
      IJ=I                                                              MUDE0110
      I1=I-1                                                            MUDE0120
      V=Z(I)                                                            MUDE0130
      DO 401 J=1,I1                                                     MUDE0140
      V=V-A(IJ)*Z(J)                                                    MUDE0150
  401 IJ=IJ+N-J                                                         MUDE0160
      W(I)=V                                                            MUDE0170
  402 Z(I)=V                                                            MUDE0180
      Z(N)=Z(N)/A(IJ)                                                   MUDE0190
```

```
      NP=N+1                                                            MUDE0200
      DO 411 NIP=2,N                                                    MUDE0210
      I=NP-NIP                                                          MUDE0220
      II=IJ-NIP                                                         MUDE0230
      V=Z(I)/A(II)                                                      MUDE0240
      IP=I+1                                                            MUDE0250
      IJ=II                                                             MUDE0260
      DO 410 J=IP,N                                                     MUDE0270
      II=II+1                                                           MUDE0280
  410 V=V-A(II)*Z(J)                                                    MUDE0290
  411 Z(I)=V                                                            MUDE0300
      RETURN                                                            MUDE0310
      END                                                               MUDE0320
      SUBROUTINE BQDMA(N,A,IA,B,BL,BU,X,O,LT,K,G)                       BQMA0010
      DIMENSION A(IA,1),B(1),BL(1),BU(1),X(1),LT(1),G(1)                BQMA0020
      IS=N                                                              BQMA0030
      IAS=N                                                             BQMA0040
      IV=N                                                              BQMA0050
      ICAC=N+N                                                          BQMA0060
      ID=ICAC                                                           BQMA0070
      DO 9 I=1,N                                                        BQMA0080
    9 G(I)=-B(I)                                                        BQMA0090
      DO 10 I=1,N                                                       BQMA0100
      X(I)=0.                                                           BQMA0110
      LT(I)=I                                                           BQMA0120
      G(ICAC+I)=A(I,I)                                                  BQMA0130
      IF(0..GE.BL(I).AND.0..LE.BU(I))GOTO10                             BQMA0140
      IF(0..LT.BL(I))X(I)=BL(I)                                         BQMA0150
      IF(0..GT.BU(I))X(I)=BU(I)                                         BQMA0160
      DO 12 J=1,I                                                       BQMA0170
   12 G(J)=G(J)+A(J,I)*X(I)                                             BQMA0180
      IF(I.EQ.N)GOTO10                                                  BQMA0190
      II=I+1                                                            BQMA0200
      DO 11 J=II,N                                                      BQMA0210
   11 G(J)=G(J)+A(I,J)*X(I)                                             BQMA0220
   10 CONTINUE                                                          BQMA0230
      K=0                                                               BQMA0240
      K1=1                                                              BQMA0250
   20 CONTINUE                                                          BQMA0260
      IOUT=0                                                            BQMA0270
      DEL=0.                                                            BQMA0280
      DO 21 I=K1,N                                                      BQMA0290
      LI=LT(I)                                                          BQMA0300
      IF( ABS(X(LI)-BL(I)).LT.1.0E-30.AND.G(I).GE.0.E0) GO TO 21
      IF( ABS(X(LI)-BU(I)).LT.1.0E-30.AND.G(I).LE.0.E0) GO TO 21
      IF(G(I).LT.0.)GOTO22                                              BQMA0330
      Z=X(LI)-BL(LI)                                                    BQMA0340
      J=1                                                               BQMA0350
      GOTO23                                                            BQMA0360
   22 CONTINUE                                                          BQMA0370
      Z=BU(LI)-X(LI)                                                    BQMA0380
      J=0                                                               BQMA0390
   23 CONTINUE                                                          BQMA0400
      IF(G(ICAC+I).LE.0.)GOTO24                                         BQMA0410
      BETA=ABS(G(I))/G(ICAC+I)                                          BQMA0420
```

```
      IF(BETA.GE.Z)GOTO24                                          BOMA0430
      Z=BETA                                                       BOMA0440
      D=.5*Z*ABS(G(I))                                             BOMA0450
      J=-1                                                         BOMA0460
      GOTO26                                                       BOMA0470
   24 CONTINUE                                                     BOMA0480
      D=Z*(ABS(G(I))-.5*Z*G(ICAC+I))                               BOMA0490
   26 CONTINUE                                                     BOMA0500
      IF(D.LT.DEL)GOTO21                                           BOMA0510
      DEL=D                                                        BOMA0520
      ALPHA=Z                                                      BOMA0530
      IOUT=I                                                       BOMA0540
      IIN=I                                                        BOMA0550
      IF(J.LT.0)IIN=0                                              BOMA0560
      LB=J                                                         BOMA0570
   21 CONTINUE                                                     BOMA0580
      IF(IOUT.NE.0)GOTO29                                          BOMA0590
   27 CONTINUE                                                     BOMA0600
      Q=0.                                                         BOMA0610
      DO 28 I=1,N                                                  BOMA0620
      LI=LT(I)                                                     BOMA0630
   28 Q=Q+X(LI)*(G(I)-B(LI))                                       BOMA0640
      Q=.5*Q                                                       BOMA0650
      RETURN                                                       BOMA0660
   29 CONTINUE                                                     BOMA0670
      SIG=1.                                                       BOMA0680
      IF(G(IOUT).GT.0.)SIG=-1.                                     BOMA0690
      LIOUT=LT(IOUT)                                               BOMA0700
      LIIN=LIOUT                                                   BOMA0710
   25 CONTINUE                                                     BOMA0720
      SAS=G(ICAC+IOUT)                                             BOMA0730
      IF(K.EQ.0)GOTO31                                             BOMA0740
      DO 30 I = 1, K
   30 G(IS+I)=G(ID+I)*A(IOUT,I)
   31 CONTINUE                                                     BOMA0770
      DO 37 I=K1,N                                                 BOMA0780
      LI=LT(I)                                                     BOMA0790
      IF(LI-LIOUT)32,37,33                                         BOMA0800
   32 Z=A(LI,LIOUT)                                                BOMA0810
      GOTO34                                                       BOMA0820
   33 Z=A(LIOUT,LI)                                                BOMA0830
   34 CONTINUE                                                     BOMA0840
      IF(K.EQ.0)GOTO36                                             BOMA0850
      DO 35 J=1,K                                                  BOMA0860
   35 Z=Z-A(I,J)*G(IS+J)                                           BOMA0870
   36 G(IS+I)=Z                                                    BOMA0880
   37 CONTINUE                                                     BOMA0890
      G(IS+IOUT)=SAS                                               BOMA0900
      IF(K.EQ.0)GOTO42                                             BOMA0910
      G(IS+K)=-A(IOUT,K)                                           BOMA0920
      IF(K.EQ.1)GOTO42                                             BOMA0930
      I=K                                                          BOMA0940
      DO 41 II=2,K                                                 BOMA0950
      I=I-1                                                        BOMA0960
      Z=-A(IOUT,I)                                                 BOMA0970
```

```
      I1=I+1                                                            BOMA0980
      DO 40 J=I1,K                                                      BOMA0990
   40 Z=Z-G(IS+J)*A(J,I)                                                BOMA1000
   41 G(IS+I)=Z                                                         BOMA1010
   42 CONTINUE                                                          BOMA1020
      IF( ABS(SIG-1.).LT.1.0E-30) GO TO 51
      DO 50 I=1,N                                                       BOMA1040
   50 G(IS+I)=-G(IS+I)                                                  BOMA1050
   51 CONTINUE                                                          BOMA1060
      IF(K.EQ.0)GOTO62                                                  BOMA1070
      DO 61 I=1,K                                                       BOMA1080
      IF( ABS(G(IS+I)).LT.1.0E-30) GO TO 61
      LI=LT(I)                                                          BOMA1100
      J=1                                                               BOMA1110
      Z=BL(LI)-X(LI)                                                    BOMA1120
      IF(G(IS+I).LT.0.)GOTO60                                           BOMA1130
      J=0                                                               BOMA1140
      Z=BU(LI)-X(LI)                                                    BOMA1150
   60 CONTINUE                                                          BOMA1160
      Z=Z/G(IS+I)                                                       BOMA1170
      IF(Z.GE.ALPHA)GOTO61                                              BOMA1180
      ALPHA=Z                                                           BOMA1190
      LB=J                                                              BOMA1200
      IIN=I                                                             BOMA1210
      LIIN=LI                                                           BOMA1220
   61 CONTINUE                                                          BOMA1230
   62 CONTINUE                                                          BOMA1240
      X(LIOUT)=X(LIOUT)+SIG*ALPHA                                       BOMA1250
      IF(K.EQ.0)GOTO71                                                  BOMA1260
      DO 70 I=1,K                                                       BOMA1270
      LI=LT(I)                                                          BOMA1280
   70 X(LI)=X(LI)+ALPHA*G(IS+I)                                         BOMA1290
   71 CONTINUE                                                          BOMA1300
      DO 72 I=K1,N                                                      BOMA1310
   72 G(I)=G(I)+ALPHA*G(IAS+I)                                          BOMA1320
      IF(IIN.EQ.0)GOTO90                                                BOMA1330
      X(LIIN)=BL(LIIN)                                                  BOMA1340
      IF(LB.EQ.0)X(LIIN)=BU(LIIN)                                       BOMA1350
      IF(IIN.EQ.IOUT)GOTO20                                             BOMA1360
      K2=K-1                                                            BOMA1370
      SG=G(ID+IIN)                                                      BOMA1380
      I1=IIN+1                                                          BOMA1390
      DO 80 I=I1,N                                                      BOMA1400
   80 G(IV+I)=A(I,IIN)                                                  BOMA1410
      IF(IIN.EQ.K)GOTO86                                                BOMA1420
      I2=IIN+2                                                          BOMA1430
      S0=1./SG                                                          BOMA1440
      DO 85 I=IIN,K2                                                    BOMA1450
      V=G(IV+I1)                                                        BOMA1460
      VD=V/G(ID+I1)                                                     BOMA1470
      S1=S0+V*VD                                                        BOMA1480
      R=S1/S0                                                           BOMA1490
      G(ID+I)=G(ID+I1)*R                                                BOMA1500
      BETA=VD/S1                                                        BOMA1510
      IF(R.GT.4.)GOTO841                                                BOMA1520
```

```
      DO 81 J=I2,N                                                    BQMA1530
   81 G(IV+J)=G(IV+J)-V*A(J,I1)                                       BQMA1540
      IF(I1.GT.K2)GOTO83                                              BQMA1550
      DO 82 J=I1,K2                                                   BQMA1560
      J1=J+1                                                          BQMA1570
   82 A(J,I)=A(J1,I1)+BETA*G(IV+J1)                                   BQMA1580
   83 CONTINUE                                                        BQMA1590
      A(K,I)=BETA                                                     BQMA1600
      DO 84 J=K1,N                                                    BQMA1610
   84 A(J,I)=A(J,I1)+BETA*G(IV+J)                                     BQMA1620
      GOTO849                                                         BQMA1630
  841 CONTINUE                                                        BQMA1640
      IF(I1.GT.K2)GOTO843                                             BQMA1650
      DO 842 J=I1,K2                                                  BQMA1660
      J1=J+1                                                          BQMA1670
  842 A(J,I)=BETA*G(IV+J1)+A(J1,I1)/R                                 BQMA1680
  843 CONTINUE                                                        BQMA1690
      A(K,I)=BETA                                                     BQMA1700
      DO 844 J=K1,N                                                   BQMA1710
  844 A(J,I)=BETA*G(IV+J)+A(J,I1)/R                                   BQMA1720
      DO 845 J=I2,N                                                   BQMA1730
  845 G(IV+J)=G(IV+J)-V*A(J,I1)                                       BQMA1740
  849 CONTINUE                                                        BQMA1750
      LT(I)=LT(I1)                                                    BQMA1760
      S0=S1                                                           BQMA1770
      I1=I2                                                           BQMA1780
   85 I2=I2+1                                                         BQMA1790
      SG=1./S1                                                        BQMA1800
      LT(K)=LIIN                                                      BQMA1810
      G(ID+K)=SG                                                      BQMA1820
      IF(IIN.EQ.1)GOTO851                                             BQMA1830
      II=IIN-1                                                        BQMA1840
      DO 852 I=1,II                                                   BQMA1850
      Z=A(IIN,I)                                                      BQMA1860
      DO 853 J=IIN,K2                                                 BQMA1870
  853 A(J,I)=A(J+1,I)                                                 BQMA1880
  852 A(K,I)=Z                                                        BQMA1890
  851 CONTINUE                                                        BQMA1900
   86 CONTINUE                                                        BQMA1910
      DO 87 I=K1,N                                                    BQMA1920
   87 G(ICAC+I)=G(ICAC+I)+SG*G(IV+I)**2                               BQMA1930
      K1=K                                                            BQMA1940
      K=K2                                                            BQMA1950
      IIN=0                                                           BQMA1960
      ALPHA=1E75                                                      BQMA1970
      SAS=G(ICAC+IOUT)                                                BQMA1980
      IF(SAS.GT.0.)ALPHA=ABS(G(IOUT))/SAS                             BQMA1990
      IF(G(IOUT).LT.0.)GOTO898                                        BQMA2000
      J=1                                                             BQMA2010
      Z=X(LIOUT)-BL(LIOUT)                                            BQMA2020
      GOTO899                                                         BQMA2030
  898 CONTINUE                                                        BQMA2040
      J=0                                                             BQMA2050
      Z=BU(LIOUT)-X(LIOUT)                                            BQMA2060
  899 CONTINUE                                                        BQMA2070
```

```
      IF(Z.GE.ALPHA)GOTO25                                              BQMA2080
      ALPHA=Z                                                           BQMA2090
      LB=J                                                              BQMA2100
      IIN=IOUT                                                          BQMA2110
      LIIN=LIOUT                                                        BQMA2120
      GOTO25                                                            BQMA2130
   90 CONTINUE                                                          BQMA2140
      K2=K1+1                                                           BQMA2150
      IF( ABS(SIG-1.).LT.1.0E-30) GO TO 91
      DO 901 I=K1,N                                                     BQMA2170
  901 G(IAS+I)=-G(IAS+I)                                                BQMA2180
   91 CONTINUE                                                          BQMA2190
      IF(IOUT.EQ.K1)GOTO97                                              BQMA2200
      LT(IOUT)=LT(K1)                                                   BQMA2210
      LT(K1)=LIOUT                                                      BQMA2220
      G(IAS+IOUT)=G(IAS+K1)                                             BQMA2230
      G(ICAC+IOUT)=G(ICAC+K1)                                           BQMA2240
      G(ICAC+K1)=SAS                                                    BQMA2250
      G(IOUT)=G(K1)                                                     BQMA2260
      IF(K.EQ.0)GOTO93                                                  BQMA2270
      DO 92 I=1,K                                                       BQMA2280
      Z=A(K1,I)                                                         BQMA2290
      A(K1,I)=A(IOUT,I)                                                 BQMA2300
   92 A(IOUT,I)=Z                                                       BQMA2310
   93 CONTINUE                                                          BQMA2320
      IF(K2.EQ.IOUT)GOTO95                                              BQMA2330
      I1=IOUT-1                                                         BQMA2340
      DO 94 I=K2,I1                                                     BQMA2350
   94 A(IOUT,I)=A(I,K1)                                                 BQMA2360
   95 CONTINUE                                                          BQMA2370
      IF(IOUT.EQ.N)GOTO97                                               BQMA2380
      I1=IOUT+1                                                         BQMA2390
      DO 96 I=I1,N                                                      BQMA2400
   96 A(I,IOUT)=A(I,K1)                                                 BQMA2410
   97 CONTINUE                                                          BQMA2420
      G(K1)=0.                                                          BQMA2430
      K=K1                                                              BQMA2440
      IF(K.EQ.N)GOTO27                                                  BQMA2450
      DO 98 I=K2,N                                                      BQMA2460
      Z=G(IAS+I)/SAS                                                    BQMA2470
      A(I,K1)=Z                                                         BQMA2480
   98 G(ICAC+I)=G(ICAC+I)-Z*G(IAS+I)                                    BQMA2490
      K1=K2                                                             BQMA2500
      GOTO20                                                            BQMA2510
      END                                                               BQMA2520
      SU BROUTINE BQDMB(N,A,IA,G,K)                                     BQMB0010
C                                                                       BQMB0020
      DIMENSION A(IA,1),G(1)                                            BQMB0030
      IF(K.EQ.0)RETURN                                                  BQMB0040
      ID=N+N                                                            BQMB0050
      G(N+1)=1./G(ID+1)                                                 BQMB0060
      IF(K.EQ.1)RETURN                                                  BQMB0070
      N1=K-1                                                            BQMB0080
      DO 111 I=1,N1                                                     BQMB0090
      I1=I+1                                                            BQMB0100
```

```
      A(I1,I)=-A(I1,I)                                                  BOMB0110
      IF(I.EQ.N1)GOTO102                                                BOMB0120
      II=I+2                                                            BOMB0130
      DO 101 J=II,K                                                     BOMB0140
      Z=A(J,I)                                                          BOMB0150
      J1=J-1                                                            BOMB0160
      DO 100 L=I1,J1                                                    BOMB0170
  100 Z=Z+A(J,L)*A(L,I)                                                 BOMB0180
  101 A(J,I)=-Z                                                         BOMB0190
  102 CONTINUE                                                          BOMB0200
      AA=1./G(ID+I1)                                                    BOMB0210
      G(N+I1)=AA                                                        BOMB0220
      DO 111 J=1,I                                                      BOMB0230
      Z=A(I1,J)*AA                                                      BOMB0240
      G(N+J)=G(N+J)+Z*A(I1,J)                                           BOMB0250
      IF(I.EQ.1)GOTO111                                                 BOMB0260
      J1=J+1                                                            BOMB0270
      DO 110 L=J1,I                                                     BOMB0280
  110 A(L,J)=A(L,J)+A(I1,L)*Z                                           BOMB0290
  111 A(I1,J)=Z                                                         BOMB0300
      RETURN                                                            BOMB0310
      END
      SUBROUTINE ALAGB(N,M,X)
      REAL X(N)
      COMMON/ALAGC/F,MM,KL,IS,MK,NU
      COMMON/ALAGD/G( 50)
      COMMON/ALAGE/C(150)
      COMMON/ALAGF/GC(25,50)
      COMMON/ALAGG/T(150)
      COMMON/SCALE/VSCAL(22),CSCAL(39),FSCAL
      COMMON/CONS/XM1,XM2,XM3,XM4,XM5,XM6,XM7,XM8,XM9,XM10,XM11,        SUM00030
     1XM12,XM13,XM14,XM15,XM16,XM17,XM18,XM19,XM20,XM21,XM22,XM23,XM24, SUM00035
     1XM25,XM26,XM27,XM28,XM29,XM30,PO,EI,EO,FC,FW,RO,VST,VBE,TSR,TSF,
     1  VD,TND,TFD,TRE,PE1,PE2,BS1,BS2,BSP,VR,RCKCK,DI,                 SUM0011
     2  DC,DK3,DK4,DK5,KS,KH,S,FR,RT,XN                                 SUM00
      X1=X (1)    *VSCAL( 1)
      X2=X (2)    *VSCAL( 2)
      X3=X (3)    *VSCAL( 3)
      X4=X (4)    *VSCAL( 4)
      X5=X (5)    *VSCAL( 5)
      X6=X (6)    *VSCAL( 6)
      X7=X (7)    *VSCAL( 7)
      X8=X (8)    *VSCAL( 8)
      X9=X (9)    *VSCAL( 9)
      X10=X (10)  *VSCAL(10)
      X11=X (11)  *VSCAL(11)
      X12=X (12)  *VSCAL(12)
      X13=X (13)  *VSCAL(13)
      X14=X (14)  *VSCAL(14)
      X15=X (15)  *VSCAL(15)
      X16=X (16)  *VSCAL(16)
      X17=X (17)  *VSCAL(17)
      X18=X (18)  *VSCAL(18)
      X19=X (19)  *VSCAL(19)
      X20=X (20)  *VSCAL(20)
```

```
      X21=X (21) *VSCAL(21)
      X22=X (22) *VSCAL(22)
C     X23=X18*X19
C     WRITE(6,8999)X1,X2,X3,X4,X17,X18,X19,X23
C8999 FORMAT('0TEST OF X VALU', 8F13.8)
 1000 Y1=DI*(X1*X1*X11+X4*X4*X12+X17*X17*X22)
      Y2=XM1*(X3*X3*X2*X2*X1+X6*X6*X5*X5*X4+X19*X19*X18*X18*X17)
      Y3=DK3*X9+DK4*X10+DK5*X21
      Y4=PO*(1./X8-1.)/KH+PO/KS/X8
      F=Y1+Y2+Y3+Y4
      PIF=(XM4/X8)**2*(X13+X14)
      PQ=(XM4/X8) *(VST+.1*VBE)+XM10*(XM8*PO*XM11/X8+XM9*XM12/X20)
      PD=XM7*VD/X8+(XM6*FR/12./XN)*(PO*XM8*XM13/X8+XM9*XM14/X20)
      POF=EO/XM6*XM7**2*RCKCK/X21+(XM16*RCKCK/X21+X16)*(PO*PO*XM8**2/X8*
     1*2+XM9**2/3./X20**2)+XM7**2*(XN*EI/XM6-2./X8)*RCKCK/X21+1./XM8/X18
     2**2*XM17*X22
    7 C(1)=PO*(1./X8-1.)-PIF-PQ-PD-POF
    8 C(2)=X13*X3*X3-XM2*X1*X2*X2
   12 C(3)=X14*X6*X6-XM2*X5*X5*X4
    9 C(4)=X9*X9+X15*X15*X9**3/X7-PE1**2*(X10*X10+(X15*X15*X9/X7)*
     1(X9-X10*XM18)**2)
      C(5)=X1*X1*X2*X2-X7*XM4/(X8*BS1)
   13 C(6)=X4*X4*X5*X5-X7*XM4/(X8*BS2*PE2)
      C(7)=XM5*X3*X2-X11*XM3+.5*X1
      C(8)=XM5*X6*X5-X12*XM3+.5*X4
    2 C(9)=X17*X17*X18*X18-(X20/BSP)*(XM8*PO/X8+XM9/X20)
      C(10)=XM5*X19*X18-X22*XM3+.5*X17
      C(11) = VR-(PO*XM8/XN/X8+XM9/XN/X20)*RCKCK/X21/EO-PO/2./XM6/FR/X21
     1/EO
      C(12)=X16*X19*X19/2.-XM2*X18*X18*X17
      C(13)=XM26*X7**2*X10/X15-XM30*X7*X10-XM27*(XM28/X8**2+XM29/X20**2)
     1**0.5
      C(14)=.97-X8
   11 C(15)=RT-X13-X14
      C(16)=X10-1.E-6
        C(17)=X9-1.E-6
C     WRITE(6,8994) (C(I), I=1,17)
C8994 FORMAT (E15.6)
      DO 103 I=1,22
  103 C(17+I)=X(I)
      G(1)=DI*X11*2.*X1+XM1*X3*X3*X2*X2
      G(2)=XM1*X3*X3*X1*2.*X2
      G(3)=XM1*X2*X2*X1*2.*X3
      G(4)=DI*X12*2.*X4+XM1*X6*X6*X5*X5
      G(5)=XM1*X6*X6*X4*2.*X5
      G(6)=XM1*X5*X5*X4*2.*X6
      G(8)=-PO/X8/X8/KH-PO/KS/X8**2
      G(9)=DK3
      G(10)=DK4
      G(12)=DI*X4*X4
      G(17)=DI*X22*2.*X17+XM1*X19*X19*X18*X18
      G(18)=XM1*X19*X19*X17*2.*X18
      G(19)=XM1*X18*X18*X17*2.*X19
      G(21)=DK5
```

```
      G(22)=DI*X17*X17
      GC(8,1)=-PO/X8/X8+(2./X8**3)*XM4*XM4*(X13+X14)+(XM4/X8**2)*(V
     1SI+.1*VBE)+XM10* XM8*XM11*PO/X8**2
     2+XM7*VD/X8**2+(XM6*FR/12./XN)*PO*XM8*XM13/X8**2
     3+2.*PO**2*XM8**2/X8**3*(XM16*RCKCK/X21+X16)-2.*XM7**2*RCKCK/
     4X21/X8**2
      GC(13,1)=-(XM4/X8)**2
      GC(14,1)=-(XM4/X8)**2
      GC(16,1)=-PO*PO*XM8**2/X8**2-XM9**2/(3.*X20*X20)
      GC(18,1)=2./XM8/X18**3*XM1 **X22
      GC(20,1)=XM10*XM9*XM12/X20**2+(XM16*RCKCK/X21+X16)*2.*(XM9**2/
     1(3.*X20**3))+(XM6*FR/12./XN*XM9*XM14/X20**2)
      GC(21,1)=EO/XM6*XM7**2*RCKCK/X21**2+XM16*RCKCK/X21**2*(PO*PO*XM8
     1**2/X8**2+XM9*XM9/3./X20**2)+XM7**2*(XN*EI/XM6-2./X8)*RCKCK/X21**2
      GC(22,1)=-1./XM8/X18**2*XM17
      GC(13,2)=X3*X3
      GC(3,2)=X13*2.*X3
      GC(2,2)=-XM2*X1*2.*X2
      GC(1,2)=-XM2*X2*X2
C
      GC(14,3)=X6*X6
      GC(6,3)=X14*2.*X6
      GC(5,3)=-XM2 *X4*2.*X5
      GC(4,3)=-XM2*X5*X5
      GC(15,4)=2.*X15*X9**3/X7-PE1*PE1*(2.*X15*X9/X7)*(X9-X10*XM18)**2
      GC(9,4)=2.*X9+3.*X9*X9*X15*X15/X7-PE1*PE1*((X15*X15*X9/X7)*2.*(X9
     1-X10*XM18)+(X9-X10*XM18)**2*X15**2/X7)
      GC(7,4)=-X15*X15*X9**3/X7/X7+PE1*PE1*(X15*X15*X9/X7/X7)*(X9-X10*
     1XM18)**2
      GC(10,4)=-PE1*PE1*(2.*X10-(X15*X15*X9/X7)*2.*(X9-X10*XM18)*XM18)
      GC(1,5)=X2*X2*2.*X1
      GC(2,5)=X1*X1*2.*X2
      GC(7,5)=-XM4/(X8*BS1)
      GC(8,5)=X7*XM4/(BS1*X8*X8)
      GC(4,6)=X5*X5*2.*X4
      GC(5,6)=X4*X4*2.*X5
      GC(7,6)=-XM4/(X8*BS2*PE2)
      GC(8,6)=X7*XM4/(PE2*BS2*X8*X8)
C
      GC(2,7)=XM5*X3
      GC(3,7)=XM5*X2
      GC(11,7)=-XM3
      GC(1,7)=.5
      GC(5,8)=XM5*X6
      GC(6,8)=XM5*X5
C
      GC(12,8)=-XM3
      GC(4,8)=.5
      GC(17,9)=X18*X18*2.*X17
      GC(18,9)=X17*X17*2.*X18
      GC(20,9)=-(XM8*PO/X8+XM9/X20)/BSP+(X20/BSP)*XM9/X20**2
      GC(8,9)=X20/BSP*XM8*PO/X8**2
      GC(18,10)=XM5*X19
      GC(19,10)=XM5*X18
      GC(22,10)=-XM3
```

```
      GC(17,10)=.5
      GC(8,11) = PO*RCKCK* XM8/(XN*X8**2*X21)/EO
      GC(21,11) = (XM8*PO/X8+XM9/X20)/XN*RCKCK/X21**2/EO+PO/(2.*XM6*FR
     1*X21**2)/EO
      GC(20,11) =  XM9*RCKCK/(XN*X20**2*X21)/EO
      GC(16,12)=X19*X19 *.5
      GC(19,12)=X16*X19
      GC(18,12)=-XM2*X17*2.*X18
      GC(17,12)=-XM2*X18*X18
      GC(7,13)=2.*XM26*X7*X10/X15-XM30*X10
      GC(8,13)=XM27*XM28/X8**3/SQRT(XM28/X8**2+XM29/X20**2)
      GC(10,13)=XM26*X7**2/X15-XM30*X7
      GC(15,13)=-XM26*X7**2*X10/X15**2
      GC(20,13)=XM27*XM29/X20**3/SQRT(XM28/X8**2+XM29/X20**2)
      GC(8,14)=-1.
      GC(13,15)=-1.
      GC(14,15)=-1.
      GC(10,16)=1.
       GC(9,17)=1.
      DO 104 I =1,N
104   GC(I,I+17)=1.0
      DO 101 I=1,N
      G(I)=G(I)*VSCAL(I)/FSCAL
101   CONTINUE
      DO 102 I=1,M
      C(I)=C(I)/CSCAL(I)
      DO 102  J=1,N
      GC(J,I)=GC(J,I)*VSCAL(J)/CSCAL(I)
102   CONTINUE
C     WRITE(6,8995) (C(I), I=1,M)
C8995 FORMAT(4E15.6)
      RETURN
      END
```

APPENDIX H

DERIVATIONS OF STATE-SPACE EQUATIONS

The objective here is to derive the state space model for the intervals T_{ON} and T_{Fl} of the continuous MMF operation of the two-winding buck/boost converter. The development will begin with the basic Kirchoff current and voltage law equations and apply matrix notation to form the constituents necessary to formulate A_1, A_2, B_1, B_2, C_1, C_2 and E_1 and E_2 used in equations (4.2.3.1) and (4.2.3.2) in the text, Volume I.

H.1 During T_{ON} (refer to Figure 2.3.2. (b)) the KVL equation on the primary side is:

$$\frac{d\phi}{dt} = \frac{v_I}{N_P} - \frac{R_P}{N_P} \; ip \qquad (H.1.1)$$

During T_{ON}, $ip = \frac{N_P}{L_P} \; \phi$

$$\therefore \quad \frac{d\phi}{dt} = \frac{v_I}{N_P} - \frac{R_P}{L_P} \; \phi \qquad (H.1.2)$$

KVL equations for the secondary side are:

$$V_C + i_1 R_C + (i_1 - i_2)RL$$

$$i_2 = -i_0 \qquad (H.1.3)$$

$$i_C = \frac{C dv_C}{dt} \quad i_1 = -i_C$$

The above three equations can be simplified to,

$$\frac{dv_C}{dt} = \frac{-1}{(R_C + R_L)C} \; v_C + \frac{R_L}{(R_C + R_L)C} \; i_O \qquad (H.1.4)$$

The output equations are of the following form:

$$v_0 = (i_1 - i_2)R_L$$

$$= \frac{R_L}{R_C + R_L} v_C + (R_C//R_L)i_0 \qquad \text{(H.1.5)}$$

$$i_p = \frac{N_P}{L_P} \phi$$

The matrix form of the state equations format $\dot{\underline{x}} = A_1\underline{x} + B_1\underline{u}$ and $\underline{y} = C_1\underline{x} + E_1\underline{u}$ is:

$$\frac{d}{dt}\begin{bmatrix}\phi \\ v_C\end{bmatrix} = \begin{bmatrix} -R_P/L_P & 0 \\ 0 & -\frac{1}{(R_C + R_L)C} \end{bmatrix} \begin{bmatrix}\phi \\ v_C\end{bmatrix} + \begin{bmatrix} 1/N_P & 0 \\ 0 & \frac{R_L}{(R_C + R_L)C} \end{bmatrix} \begin{bmatrix} v_I \\ i_0 \end{bmatrix} \qquad \text{(H.1.6)}$$

$$\begin{bmatrix} v_0 \\ i_p \end{bmatrix} = \begin{bmatrix} 0 & \frac{R_L}{R_C + R_L} \\ N_P/L_P & 0 \end{bmatrix} \begin{bmatrix}\phi \\ v_C\end{bmatrix} + \begin{bmatrix} 0 & R_C//R_L \\ 0 & 0 \end{bmatrix} \begin{bmatrix} v_I \\ i_0 \end{bmatrix} \qquad \text{(H.1.7)}$$

H.2 During T_{F1} (refer to Figure 4.2.3.2.(c)) KVL equation for i_1, i_2 and i_3:

$$v_S = i_1 R_S + (i_1 - i_2)R_C + v_C \tag{H.2.1}$$

$$v_C = (i_2 - i_1)R_C + (i_2 - i_3)R_L \tag{H.2.2}$$

$$i_3 = -i_0 \tag{H.2.3}$$

Simplifying equations (H.2.1) through (H.2.3) results in the following:

$$v_S = (R_S - R_C)i_1 - \frac{R_C}{R_C + R_L}\,[V_C + R_C i_1 - R_L i_0] + v_C \tag{H.2.4}$$

$$i_2 \triangleq \frac{v_C + R_C i_1 - R_L i_0}{R_C + R_L} \tag{H.2.5}$$

From Figure 4.2.3.2(c) the following equations can be derived:

$$i_1 = i_S = \frac{N_S}{L_S}\,\phi \tag{H.2.6}$$

$$v_S = -N_S \frac{d\phi}{dt}$$

$$\frac{d\phi}{dt} = \frac{-[R_S + R_C//R_L]}{L_S}\,\phi - \frac{R_L}{R_C + R_L} \cdot \frac{1}{N_S}\,v_C - \frac{R_C//R_L}{N_S}\,i_0 \tag{H.2.8}$$

From Figure 4.2.3.2(c) it can be seen that

$$i_C = i_1 - i_2 = C\frac{dv_C}{dt} \quad \text{and} \quad i_1 = \frac{N_S}{L_S}\,\phi \tag{H.2.9}$$

Substituting equation (H.2.8) into (H.2.9),

$$\frac{dv_C}{dt} = \frac{N_S}{CL_S}\,\frac{R_L}{R_C + R_L}\,\phi - \frac{1}{(R_C + R_L)C}\,v_C + \frac{R_L}{(R_C + P_L)C}\,i_0 \tag{H.2.10}$$

Output:

$$v_0 = R_L(i_2 - i_3)$$

$$= \frac{R_L}{R_C + R_L} v_C + (R_C//R_L)\frac{N_S}{L_S}\phi + (R_C//R_L)i_0 \qquad \text{(H.2.11)}$$

In matrix form, $\dot{\underline{x}} = A_2\underline{x} + B_2\underline{u}$ and $\underline{y} = C_2\underline{x} + E_2\underline{u}$ are expressed in the following form:

$$\frac{d}{dt}\begin{bmatrix}\phi \\ v_C\end{bmatrix} = \begin{bmatrix} -\left(\frac{R_S + R_C//R_L}{L_S}\right) & -\frac{R_L}{R_C + R_L}\cdot\frac{1}{N_S} \\ \frac{R_L}{R_C + R_L}\cdot\frac{N_S}{L_S C} & -\frac{1}{R_C + R_L}\cdot\frac{1}{C}\end{bmatrix}\begin{bmatrix}\phi \\ v_C\end{bmatrix} +$$

$$\begin{bmatrix} 0 & -R_C//R_L\cdot\frac{1}{N_S} \\ 0 & \frac{R_L}{R_C + R_L}\cdot 1/C\end{bmatrix}\begin{bmatrix}v_I \\ i_0\end{bmatrix} \qquad \text{(H.2.12)}$$

$$\begin{bmatrix}v_o \\ i_p\end{bmatrix} = \begin{bmatrix} R_C//R_L\cdot\frac{N_S}{L_S} & \frac{R_L}{R_C + R_L} \\ 0 & 0\end{bmatrix}\begin{bmatrix}\phi \\ v_C\end{bmatrix} +$$

$$\begin{bmatrix} 0 & R_C//R_L \\ 0 & 0\end{bmatrix}\begin{bmatrix}v_I \\ i_0\end{bmatrix} \qquad \text{(H.2.13)}$$

APPENDIX I

DERIVATIONS OF TRANSFER FUNCTIONS FOR THE SMALL SIGNAL MODEL

In Appendix B a detailed formulation for the steady-state (dc) transfer function, the input-to-output transfer function, and the duty cycle-to-output transfer function is presented.

I.1 Derivation for steady state (dc) input-to-output relation

$$\underline{Y} = \begin{bmatrix} V_0 \\ I_p \end{bmatrix} = (E - CA^{-1}B)\underline{U}$$

$$= \frac{1}{\omega_0^2 \left[\frac{R_S + D'R_C}{D'^2 R_L}\right]} \left\{ \begin{bmatrix} 0 & R_C \\ 0 & 0 \end{bmatrix} - \begin{bmatrix} \frac{N_S R_C}{L_e D'} & 1 \\ \frac{DN_P}{L_P} & 0 \end{bmatrix} \right.$$

$$\left. \begin{bmatrix} -1/R_L C & \frac{D'}{N_S} \\ \frac{-N_S \omega_0^2}{D'} & \frac{-R_S - D'R_C}{L_e D'^2} \end{bmatrix} \begin{bmatrix} \frac{D}{N_P} & \frac{-D'R_C}{N_S} \\ 0 & \frac{1}{C} \end{bmatrix} \right\} \begin{bmatrix} V_I \\ 0 \end{bmatrix} \qquad (I.1.1)$$

$$Y = \frac{1}{\omega_0^2 \left(\frac{R_S + D'R_C}{D'^2 R_L} + 1 \right)} \cdot$$

$$\begin{bmatrix} \frac{DN_S R_C \omega_0^2}{D'N_P R_L} + \frac{N_S D \omega_0^2}{N_P D'} & R_C \omega_0^2 \left(\frac{1 + R_S + D'R_C}{D'^2 R_L} \right) - \frac{R_C^2 \omega_0^2}{R_L} - 2R_C \omega_0^2 + \left(\frac{R_S + D'R_C}{D'^2} \right) \omega_0^2 \\ \frac{D^2}{L_P R_L C} & \frac{DD'N_P}{L_P N_S C} \left(\frac{R_C}{R_L} + 1 \right) \end{bmatrix}$$

$$\begin{bmatrix} V_I \\ 0 \end{bmatrix} \qquad \text{(I.1.2)}$$

$$= \frac{1}{\omega_0^2 \left(\frac{R_S + D'R_C}{D'^2 R_L} + 1 \right)} \begin{bmatrix} \frac{DN_S \omega_0^2}{D'N_P} \left(\frac{R_C}{R_L} + 1 \right) \\ \frac{D^2}{L_P R_L C} \end{bmatrix} V_I \qquad \text{(I.1.3)}$$

Assume

$$\frac{R_S + D'R_C}{D'^2 R_L} \ll 1 \qquad \frac{R_C}{R_L} \ll 1$$

$$\therefore \quad \underline{Y} = \begin{bmatrix} V_O \\ I_P \end{bmatrix} = \begin{bmatrix} \frac{DN_S}{D'N_P} \\ \frac{D^2}{\omega_0^2 L_P R_L C} \end{bmatrix} V_I = \begin{bmatrix} \frac{DN_S}{D'N_P} \\ \frac{D^2 L_e}{L_P R_L} \end{bmatrix} V_I \qquad \text{(I.1.4)}$$

I.2 Derivation of the input-to-output transfer function.

$$\hat{\underline{y}}(s) = [C[SI-A]^{-1}B + E]\hat{\underline{u}}(s)$$

$$[SI-A] = \begin{bmatrix} S + \frac{R_S + D'R_C}{L_e D'^2} & \frac{D'}{N_S} \\ \frac{-N_S \omega_0^2}{D'} & S + \frac{1}{R_L C} \end{bmatrix} \qquad (I.2.1)$$

$$[SI-A]^{-1} = \frac{1}{S^2 + \left(\frac{R_S + D'R_C}{L_e D'^2} + \frac{1}{R_L C}\right) S + \omega_0^2 \left(\frac{R_S + D'R_C}{R_L D'^2} + 1\right)}$$

$$\begin{bmatrix} S + \frac{1}{R_L C} & \frac{-D'}{N_S} \\ \frac{N_S \omega_0^2}{D'} & S + \frac{R_S + D'R_C}{L_e D'^2} \end{bmatrix} \qquad (I.2.2)$$

$$\frac{R_S + D'R_C}{R_L D'^2} \ll 1$$

$$[SI-A]^{-1} = \frac{1}{S^2 + 2\zeta\omega_0 + \omega_0^2} \begin{bmatrix} S + \frac{1}{R_L C} & \frac{-D'}{N_S} \\ \frac{N_S \omega_0^2}{D'} & S + \frac{R_S + D'R_C}{L_e D'^2} \end{bmatrix} \qquad (I.2.3)$$

where $\zeta = \frac{\omega_0}{2} \left[\frac{L_e}{R_L} + \left(R_e + \frac{R_C}{D'}\right) C\right]$.

$$\begin{bmatrix} \hat{v}_0(S) \\ \hat{i}_p(S) \end{bmatrix} = [C[SI-A]^{-1} B + E] \begin{bmatrix} \hat{v}_i(S) \\ \hat{i}_0(S) \end{bmatrix} \qquad \text{(I.2.4)}$$

$$= \left\{ \frac{1}{S^2 + 2\zeta\omega_o S + \omega_0^2} \begin{bmatrix} \frac{N_S R_C}{L_e D'} & 1 \\ \frac{DN_P}{L_P} & 0 \end{bmatrix} \begin{bmatrix} S + \frac{1}{R_L C} & -\frac{D'}{N_S} \\ \frac{N_S \omega_0^2}{D'} & S + \frac{R_S + D'R_C}{L_e D'^2} \end{bmatrix} \right.$$

$$\left. \begin{bmatrix} \frac{D}{N_P} & -\frac{D'R_C}{N_S} \\ 0 & \frac{1}{C} \end{bmatrix} + \begin{bmatrix} 0 & R_C \\ 0 & 0 \end{bmatrix} \right\} \begin{bmatrix} \hat{v}_i(S) \\ \hat{i}_0(S) \end{bmatrix} \qquad \text{(I.2.5)}$$

$$= \left\{ \frac{1}{S^2 + 2\zeta\omega_o S + \omega_0^2} \begin{bmatrix} \frac{N_S R_C}{L_e D'} & 1 \\ \frac{DN_P}{L_P} & 0 \end{bmatrix} \right.$$

$$\begin{bmatrix} \frac{D}{N_P}\left(S + \frac{1}{R_L C}\right) & -\frac{D'}{N_S}\left[R_C\left(S + \frac{1}{R_L C}\right) + \frac{1}{C}\right] \\ \frac{N_S D \omega_0^2}{N_P D'} & \omega_0^2\left(-R_C + \frac{R_S + D'R_C}{D'^2}\right) + \frac{S}{C} \end{bmatrix}$$

$$\left. + \begin{bmatrix} 0 & R_C \\ 0 & 0 \end{bmatrix} \right\} \begin{bmatrix} \hat{v}_i(S) \\ \hat{i}_0(S) \end{bmatrix} \qquad \text{(I.2.6)}$$

$$= \frac{1}{S^2 + 2\zeta\omega_0 S + \omega_0^2}$$

$$\begin{bmatrix} \frac{N_S D\omega_0^2}{N_P D'} (R_C CS + 1) & R_C\left[S^2 + 2\zeta\omega_o + \frac{1}{R_C C} - \frac{R_C}{L_e} S + \omega_o^2\left(\frac{R_e}{R_L} + \frac{D}{D'}\right)\right] \\ \frac{D^2}{L_P} (S + \frac{1}{R_L C}) & \frac{-DD'N_P}{L_P N_S C} (R_C CS + 1) \end{bmatrix}$$

$$\begin{bmatrix} \hat{v}_i (S) \\ \hat{i}_0 (S) \end{bmatrix} \qquad \text{(I.2.7)}$$

$$= \frac{1}{S^2 + 2\zeta\omega_0 S + \omega_0^2} \begin{bmatrix} \frac{N_S D\omega_0^2}{N_P D'} (R_C CS + 1) & R_C\left[S^2 + z_1 S + \omega_o^2 z_2\right] \\ \frac{D^2}{L_P} (S + \frac{1}{R_L C}) & \frac{-DN_S}{D'N_P} \omega_o^2 (R_C CS + 1) \end{bmatrix}$$

$$\begin{bmatrix} \hat{v}_i (S) \\ \hat{i}_0 (S) \end{bmatrix} \qquad \text{(I.2.8)}$$

$$z_1 = 2\zeta\omega_o + \frac{1}{R_C C} - \frac{R_C}{L_e}$$

$$z_2 = \frac{R_e}{R_L} + \frac{D}{D'}$$

.3 Derivation of the duty cycle-to-output transfer function,

$$\frac{\hat{y}(S)}{\hat{d}(S)} = C(SI-A)^{-1}\left[(A_1-A_2)\underline{X} + (B_1-B_2)\underline{U}\right] + \left[(C_1-C_2)\underline{X} + (E_1-E_2)\underline{U}\right] \qquad \text{(I.3.1)}$$

$$\underline{X} = -A^{-1}B\underline{U}$$

$$= \frac{-1}{\omega_0^2\left(1 + \frac{R_S + D'R_C}{D'^2R_L}\right)} \begin{bmatrix} -\frac{1}{R_LC} & \frac{D'}{N_S} \\ \frac{-N_S\omega_0^2}{D'} & \frac{-R_S-D'R_C}{L_eD'^2} \end{bmatrix} \begin{bmatrix} \frac{D}{N_P} & \frac{-D'R_C}{N_S} \\ 0 & \frac{1}{C} \end{bmatrix} \begin{bmatrix} V_I \\ 0 \end{bmatrix} \qquad \text{(I.3.2)}$$

$$= \begin{bmatrix} \frac{D}{N_PR_LC} \\ \frac{N_SD\omega_0^2}{N_PD'} \end{bmatrix} \frac{V_I}{\omega_0^2\left(1 + \frac{R_S + D'R_C}{D'^2R_L}\right)} \qquad \text{(I.3.3)}$$

$$\frac{R_S + D'R_C}{D'^2R_L} << 1$$

$$X = \begin{bmatrix} \frac{DL_e}{R_LN_P} \\ \frac{N_SD}{N_PD'} \end{bmatrix} \quad V_I = \begin{bmatrix} \frac{D'L_e}{N_sR_L} \\ 1 \end{bmatrix} V_0 \qquad \text{(I.3.4)}$$

$$(A_1 - A_2)\underline{X} + (B_1 - B_2)\underline{U} =$$

$$= \begin{bmatrix} -\frac{R_P}{L_P} & +\frac{R_S + R_C//R_L}{L_S} & \frac{R_L}{R_C + R_L} & \frac{1}{N_S} \\ \frac{-R_L}{R_C + R_L} & \frac{N_S}{L_S C} & 0 & \end{bmatrix} \begin{bmatrix} \frac{D'L_e}{N_S R_L} \\ 1 \end{bmatrix} V_0 +$$

$$\begin{bmatrix} \frac{1}{N_P} & \frac{R_C}{N_S} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \frac{D'N_P}{DN_S} & V_0 \\ 0 \end{bmatrix} \qquad (I.3.5)$$

$$= \begin{bmatrix} \frac{DR_C + D'R_L\ (D-D')}{DD'N_S R_L} \\ \frac{-1}{D'R_L C} \end{bmatrix} V_0 \simeq \begin{bmatrix} \frac{1}{N_S D} \\ \frac{-1}{D'R_L C} \end{bmatrix} V_0 \qquad (I.3.6)$$

$$(C_1 - C_2)\underline{X} + (E_1 - E_2)\underline{U} = (C_1 - C_2)\underline{X}$$

$$= \begin{bmatrix} -\frac{R_C N_S}{L_S} & 0 \\ \frac{N_P}{L_P} & 0 \end{bmatrix} \begin{bmatrix} \frac{D'L_e}{N_S R_L} \\ 1 \end{bmatrix} V_0 \qquad (I.3.7)$$

$$= \begin{bmatrix} -\frac{R_C}{R_L D'} \\ \frac{N_P L_S}{N_S L_P R_C D'} \end{bmatrix} V_0 \qquad (I.3.8)$$

$$\frac{L_S}{L_P} = \frac{N_S^2}{N_P^2}$$

$$(C_1 - C_2)\underline{X} = \begin{bmatrix} -\frac{R_C}{R_L D'} \\ \\ \frac{N_S}{N_P R_L D'} \end{bmatrix} V_0 \qquad \text{(I.3.9)}$$

$$\frac{\hat{\underline{y}}(S)}{\hat{d}(S)} = C(SI-A)^{-1}\left[(A_1-A_2)\underline{X} + (B_1-B_2)\underline{U}\right] + \left[(C_1-C_2)\underline{X} + (E_1-E_2)\underline{U}\right]$$

$$= C(SI-A)^{-1}\left[(A_1-A_2)X + (B_1-B_2)\underline{U}\right] + (C_1-C_2)\underline{X} \qquad \text{(I.3.10)}$$

$$= \frac{1}{S^2 + 2\zeta\omega_0 S + \omega_0^2} \begin{bmatrix} \frac{N_S R_C}{L_e D'} & 1 \\ \frac{DN_P}{L_P} & 0 \end{bmatrix} \begin{bmatrix} S + \frac{1}{R_L C} & \frac{-D'}{N_S} \\ \frac{N_S \omega_0^2}{D'} & S + \frac{R_S + D' R_C}{L_e D'^2} \end{bmatrix}$$

$$\begin{bmatrix} \frac{1}{DN_S} \\ \\ \frac{-1}{D' R_L C} \end{bmatrix} V_0 + \begin{bmatrix} -\frac{R_C}{R_L D'} \\ \\ \frac{N_S}{N_P R_L D'} \end{bmatrix} V_0 \qquad \text{(I.3.11)}$$

$$= \frac{1}{S^2 + 2\zeta\omega_0 S + \omega_0^2} \begin{bmatrix} \frac{N_S R_C}{L_e D'} & 1 \\ \frac{DN_P}{L_P} & 0 \end{bmatrix} \begin{bmatrix} \frac{1}{N_S D}\left(S + \frac{D+1}{R_L C}\right) \\ -\frac{1}{D'}\left(\frac{S}{R_L C} + \omega_0^2\left[\frac{R_e + R_C/D'}{R_L} - \frac{1}{D}\right]\right) \end{bmatrix} V_0$$

$$+ \begin{bmatrix} -\frac{R_C}{R_L D'} \\ \\ \frac{N_S}{N_P R_L D'} \end{bmatrix} V_0 \qquad \text{(I.3.12)}$$

I-3

$$= \begin{bmatrix} \frac{\omega_0^2}{DD'}(R_C CS+1) - \frac{\omega_0^2}{D'R_L}(L_e S+R_e+\frac{D}{D'}R_C) \\ \frac{N_P}{N_S L_P R_L C}(R_L CS + (D+1)) \end{bmatrix} \frac{V_0}{s^2 + 2\zeta\omega_0 S + \omega_0^2}$$

$$+ \begin{bmatrix} -\frac{R_C}{R_L D'} \\ \frac{N_P L_S}{N_S L_P R_L D'} \end{bmatrix} V_0 \frac{s^2 + 2\zeta\omega_0 S + \omega_0^2}{s^2 + 2\zeta\omega_0 S + \omega_0^2} \qquad \text{(I.3.13)}$$

$$= \begin{bmatrix} \frac{\omega_0^2(R_C CS + 1)}{s^2 + 2\zeta\omega_0 S + \omega_0^2} \cdot \frac{V_0}{DD'}\left[1 - \frac{D}{R_L(R_C CS+1)}\left(\frac{R_C}{\omega_0^2}s^2 + \left(\frac{2R_C\zeta}{\omega_0} + L_e\right)S + R_C + \frac{R_C}{D'}\right)\right] \\ \frac{V_0}{R_L}\left\{\frac{N_P}{N_S}\left(\frac{N_S}{N_P}\right)^2 \frac{1}{L_S C} \frac{(R_C CS + D + 1)}{s^2 + 2\zeta\omega_0 S + \omega_0^2} + \frac{N_P}{N_S}\left(\frac{N_S}{N_P}\right)^2 \frac{1}{D'}\right\} \end{bmatrix} \qquad \text{(I.3.14)}$$

$$= \begin{bmatrix} \frac{\omega_0^2 V_0}{DD'} \frac{(R_C CS+1)}{s^2 + 2\zeta\omega_0 S + \omega_0^2}\left[1 - \frac{D}{R_L}\left(\frac{S}{\omega_0^2 C} + R_e + \frac{R_C}{D'}\right)\right] \\ \frac{V_0 N_S}{R_L N_P}\left\{\frac{1}{L_S C} \frac{(R_L CS + D + 1)}{s^2 + 2\zeta\omega_0 S + \omega_0^2} + \frac{1}{D'}\right\} \end{bmatrix} \qquad \text{(I.3.15)}$$

APPENDIX J

COMPUTER PROGRAM FOR DESIGN OPTIMIZATION CALCULATIONS

ORIGINAL PAGE IS
OF POOR QUALITY

```
500 FOR T=1 TO 26
2000 DATA  25,5E3,270,3000,.5
2010 DATA 18.8E-6,4,6,1.8 ,1.2,1.2,3.1416
2020 DATA .35E-6,.35E-6,.15E-6,.15E-6,1.728E-8,2,.35
2030 DATA .35,42,.9,  1,3,5,8.9E6,7.8E6,10
2100 DATA 0,0,0,0,0
2110 DATA 0,0,0,0,0,0,0
2120 DATA 0,0,0,0,0,0,0
2130 DATA 0,0,0,0,0,0,0,0,0
2200 DATA VI,F,VO,PO,T
2210 DATA G,IAC,RI,VS,VBE,VD,PI
2220 DATA TSR,TSF,TDR,TDF,RHO,FC,FW
2230 DATA BDC,K,M,VK,KCAF,KMEC,DC,DI,KTHE
2300 DATA "VSTOL"
2400 DATA L,C,RC,IP,N,A,Z,U,AC,PL,PLC,PLI,PM,PT,PD,PC,P,NB,VB,UM,WP,WC,W
 B,WCON,WT,AR
3000 Y0=1/A6+A0/A1*(1-A0/1.07/A2)
3002 Y1=(1.07*A3*A5/A0+1/(2*A1)*(1-A0/1.07/A2)*(A0*A5/Y0+A3/A2))/A7
3004 Y2=A5/Y1
3006 Y3=1-(A8+.1*A9)/A0-(A2+B0-A8)*(B2+B3+B4+B5)*A1/6/A0-A3*Y2/A2/A0
3008 Y4=4.624*B6*B7*A3/A0*A3/A0/Y3
3010 Y5=Y0/B9*(1.07*A3/A0+A0/2/Y0/A1*(1-A0/1.07/A2))
3012 Y6=.165*A1^.6*(A2-A0)*A0/Y3/A2/Y5
3014 Y7=((A3/A0-A3/A2)*(A8+.1*A9)+A3*B0/A2+.015*A3)/Y3
3016 Y8=Y7+(A3/6/A0*(B2+B3+B4+B5)*A1*(A2+B0-A8))/Y3
3018 Y9=Y8+(A3/A2+A3/A2*(A2/A0-1)*Y2)/Y3
3020 X2=(C0*A4*A3/(C1*C2*(1-.015*C1/C2)))*(A0/C2/(1-.015*C1/C2))^0.02
3021 Z0=X2+C3*Y1+C4*A3
3022 Z1=C0*A4/(C1*C2*(1-.015*C1/C2))+C7
3024 Z2=12*B7*C5*(4*B7*C5-(C6+Z1*Y6)/B8)
3026 Z3=Z1*Y4*((C6+Z1*Y6)/B8-24*B7*C5)
3028 Z4=3*Z1*Y4*Z1*Y4
3030 Z5=((-Z3-(Z3*Z3-4*Z2*Z4)^.5)/(2*Z2))^.5
3032 Z6=(Z1*Y4/Z5/Z5-4*B7*C5)* B8/(C6+Z1*Y6)
3034 Z7=1.07*A3/A0+A0/2/Y0/A1*(1-A0/1.07/A2)
3036 Z8=1/Z6*(B1*Y0*Z7*B8/B9/Z5)^.5
3038 Z9=Z6*(Y0*Z7*Z5/B1/B9/B8)^.5
3040 K0=2*B1*(Z6^.5/2+1/Z6^.5)*(Y0*Z7*Z5/B1/B9/B8)^.25
3042 K1=Y0*K0/(Z8*Z8*Z9)/(4*B1*1E-7)
3044 K2=Y4*Z8*Z9^.5/Z5+Y6*Z9*K0+Y9
3045 K9=((A3+K2)/A0)^2+4*B6*B7*Z8*Z9^.5/Z5
3046 K3=(A0/C2)*1/(1-.015*C1/C2)
3047 X0=(A2-A0)*A3*A0/A2/(A3+K2)/A1/Y5
3048 K4=A0+.015*A0*C1/(C2*(1-.015*C1/C2))
3049 X1=2*X0*.089*A1^.6*Z9*K0*A1
3050 K5=4*B7*C5*Z8*Z5*Z9^.5+C6*Z9*K0
3051 K6=C4*A3+C7*K2
3052 K7=C3*Y1*1E6
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
3053 K8=Z0+Z1*K2+K5
3054 X2=K9+X1
3055 X8=A3+K2+(A3+K2)*.015*C1/(C2*(1-.015*C1/C2))
3056 X3=(A3+K2)/A0
3058 X4=((1-(1/X3)*A3/A2)*(A8+.1*A9)+(A2+B0-A8)*(B2+B3)*A1/6)*X3
3060 X5=A3*B0/A2+(A2+B0-A8)*(B4+B5)*A1*X3/6
3062 X6=A3/A2*A3/A2*Y2*((A2/A3)*X3-1)
3064 X7=.015*A3
3066 Y(1)=Y0
3068 Y(2)=Y1
3070 Y(3)=Y2
3072 Y(4)=Z7
3073 Y(5)=Z8
3074 Y(6)=Z9
3076 Y(7)=K0
3078 Y(8)=K1
3080 Y(9)=Z5
3082 Y(10)=K2
3086 Y(11)=K9
3088 Y(12)=X1
3090 Y(13)=X2
3092 Y(14)=X4
3094 Y(15)=X5
3096 Y(16)=X6
3098 Y(17)=X7
3100 Y(18)=K3
3102 Y(19)=K4
3104 Y(20)=K5
3106 Y(21)=K6
3108 Y(22)=K7
3110 Y(23)=C0*A4*X8/C1/C2/(1-.015*C1/C2)*K3^.02
3111 Y(24)=K5+K6+K7
3112 Y(25)=Y(23)+Y(24)
3114 Y(26)=X8*A4/A0/C1
3200 X=Y(I)

E GET,RSSWC,ACCOUNT=MB53238
E R_GET.VSTOL
 NO FILE NAMES SPECIFIED.
E BASIC
$ LOAD.VSTOL
END LOAD
$ LOAD.RSSWC
END LOAD
$ RUN
WC OR RSS ; ARG ; V/V OR DB ; AC,DC OR TR; SENS.'Y OR N , %OR LIM.(P OR
L)?
? W,28,V.D,N,L
WORST CASE ANALYSIS
```

APPENDIX K

JUSTIFICATION FOR INPUT PARAMETERS

In this appendix, justification for numerical values of all input parameters are given. These parameters have been defined in chapter 5 (Volume I).

Capacitor Time Constant G

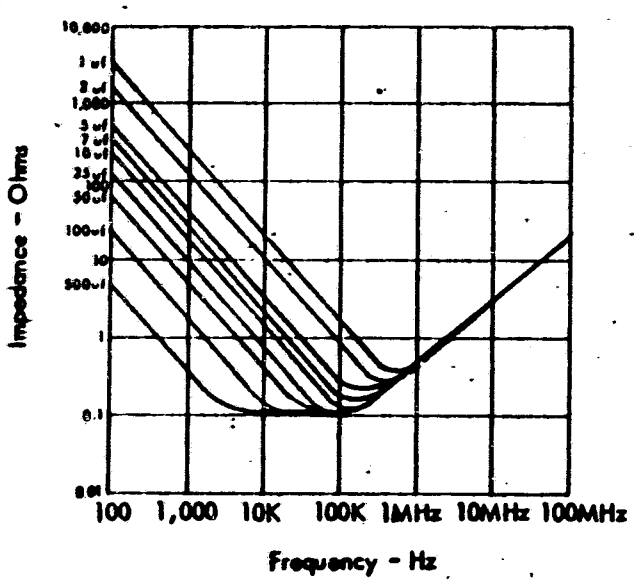
Numerically, the time constant C*RC caused by capacitance C and equivalent series resistance RC is assigned a value of 18.8E-6. This value is arrived by assuming the use of three 150V, 47 μF foil-tantalum capacitors in series to provide the necessary derating required for a 270VDC working voltage. Since each capacitor can be expected to have a maximum internal resistance of 0.4 ohms at the cold temperature of -30°C, the time constant G becomes 0.4 * 47E-6 = 18.8E-6. See Figure K1 for the supporting data, which is taken from TRW's Electronic Component Handbook.

Battery Current Ripple IAC

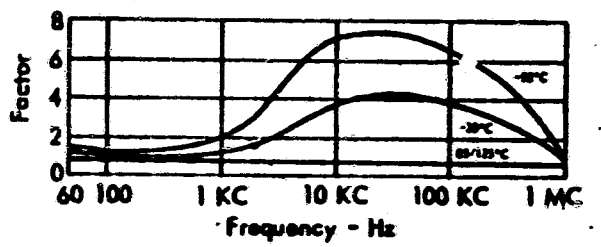
Being of emergency use only, the source EMI in the battery is not expected to be a serious concern in the power system. For the time being, a peak-to-peak 4-ampere limit is assigned arbitrarily as the ac component in the source. The limit may be adjusted to a different numerical value pending further system definitions.

Output Voltage Ripple RI

The generator specification carries a 12V peak-to-peak limit (section 3.7.5.3, NADC-VT-TS-7502). This specification is probably meant for lower frequencies corresponding to the speed range of the generator. A 6V peak-to-peak specification is assumed for the boost converter output at the converter switching frequencies. Again, this limit may be adjusted to a different value pending further system definitions.



Impedance Curves for 1A008 Capacitors at 25°C



Tantalum Foil Impedance Correction Factors for Capacitance 2 - 50 µF

Figure K1 Supporting Data for Capacitor Time Constant

Transistor Conduction Voltage Drop (VS) and Base-Transmitter Voltage (VBE)

Since the voltage across the transistor is at least 270V during its off time, and the current through it can be as high as 50A or more, presently available transistors compatible for this application will all have high conduction drop. Transistor D60T with a 400V, 200A rating is specified to have a 1.25V maximum drop at the rated current. In practical application the actual conduction drop is closely related to the converter input voltage, and the speed with which the transistor is capable of turning fully on, which in turn is a function of the transistor drive scheme. In the computer program, a conservative 1.8V is assigned to VS. This number can be reduced if techniques of connecting in series two transistors with lower voltage ratings can be utilized. The choice of VBE=1.2V is reasonable, as it is the maximum VBE for a 5-ampere base current.

Diode Voltage (VD)

Diode SA8466 by Semtech rated at 35A and 1000V appears to be more than adequate for this application. At 20-ampere the diode is measured to have a voltage drop of 1.2V. Voltage VD is therefore assigned accordingly.

Switching Times For Transistors and Diodes TSR, TSF, TDR, TDF

The switching times will be functions of switching current and base dirve. The rise and fall times TR and TF for both the transistor and the diode are assigned reasonable, conservative numbers. They are 0.35 microseconds each for the transistor, and 0.15 microseconds each for the diode.

Inductor-Related Parameters RHO, FC, FW, BDC

Due to the large flux excursion in the inductor, a powder-core is first assumed for the application, with a peak operating flux density BDC of 0.35 weber/meter^2. Copper conductors result in a resistivity RHO of 1.728E-6 ohm-meter. For a large inductor such as this, experiences would generally dictate a winding pitch factor FC=2 and a window winding fill factor FW=0.35. A BDC of 1.2 weber/m^2 is then used to make another set of runs to assess its impact to system weight.

Battery Weight/Ampere-Hour Constant (K)

The relationship between an individual cell weight and its ampere-hour capacity can be found from "Nickel-Cadmium Battery Application Engineering Handbook", 2nd Edition, General Electric Company, Publication Number GET-3148A. After plotting weight versus ampere-hour, it is seen to be essentially a straight line, with a slope of approximately 42 grams/ampere-hour, as shown in Figure K2.

Cell Capacity As A Function of Discharge Rate, (M)

The effect of discharge rate on cell capacity is shown in Figure K3, which is obtained from the aforementioned GE Handbook. A 2C discharge is estimated to produce a reduction of the cell to 90% of its ampere-hour capacity rated at 1C discharge. Therefore, M is taken as 0.9.

Cell Voltage (Vk)

Near the end of discharge of its rated ampere-hour, the cell voltage is reduced from 1.25V nominal to a lower value that is a function of temperature and discharge rate. In this application, a conservative (1V) is assumed for Vk.

Mechanical Structure Constant KMEC and Thermal Design Constant (KTHE)

These constants are important to the analysis, as they affect significantly the total system weight. Admittedly, the estimate of these constants is rather difficult, as it depends on various parameters including the vibration requirement and the thermal design at the system level. For the time being, the converter mechanical packaging weight are modelled to be functions of output power PO and total loss PL, in the form of KMEC*PO + KTHE*PL. Relying on experience primarily, KMEC and KTHE are set for this application to 5 and 10 grams/watt, respectively.

GRAPH PAPER GRAPHIC CONTROLS CORPORATION Made in New York
Printed in U.S.A.
SQUARE 10 X 10 TO THE INCH AS-0000-01

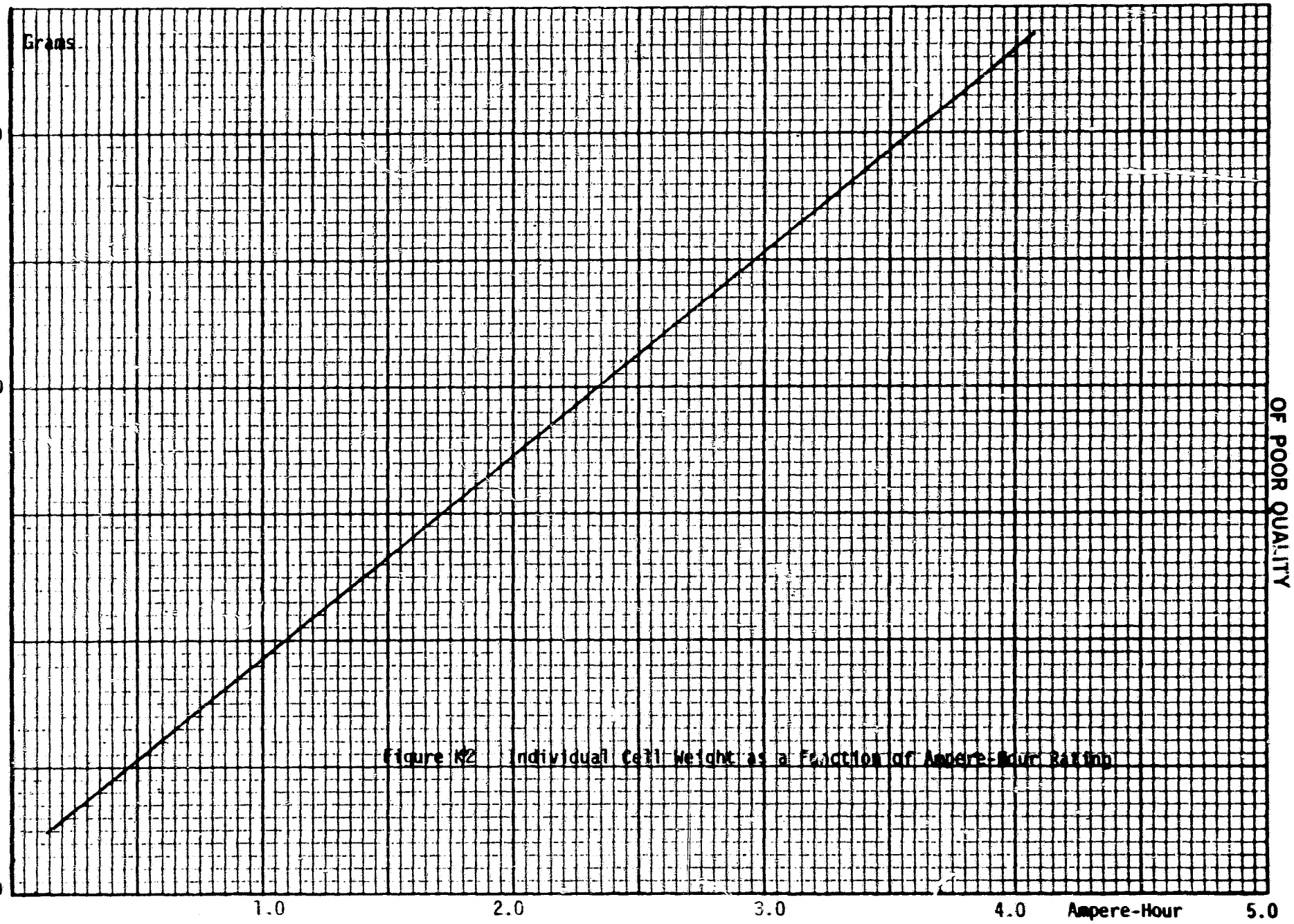


Figure K2 Individual Cell Weight as a Function of Ampere-Hour Rating

ORIGINAL PAGE IS
OF POOR QUALITY

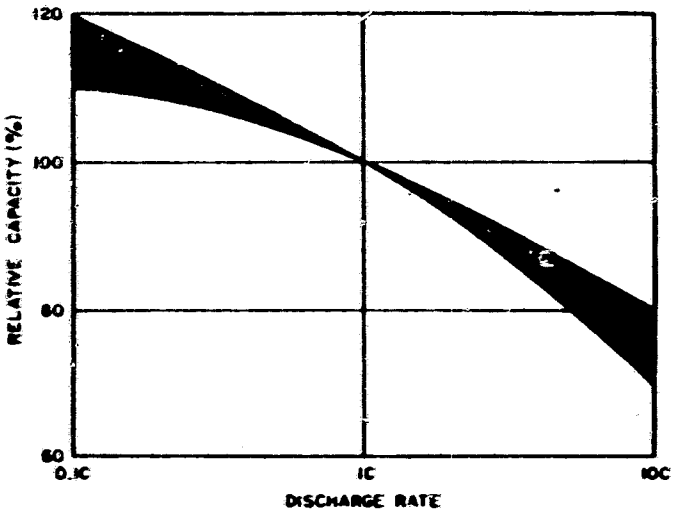


Figure K3 Cell Capacity as a Function of Discharge Rate

Capacitor Density in Grams per Microfarad (KCAP)

As previously stated, three 150-volt, 47-microfarad capacitors are connected in series to meet the required voltage with ample derating. Each capacitor weighs 17.5 grams; a total of 3 x 17.5 grams is needed to make 47/3 microfarads. In terms of grams per microfarad, the number is therefore $9 \times 17.5/47 = 3$.

Conductor Density DC and Core Density DI

The copper density is used for DC at 8.9E6 grams/meter^3. The iron density is used for DI at 7.8E6 grams/meter^3.